Word Processing for Technical Writers

Edited by:
Robert Krull

T
11
.W67
1988

Baywood's Technical Communications Series 3

Baywood Publishing Company, Inc.

Word Processing for Technical Writers

Edited by:
Robert Krull

Baywood's Technical Communications Series:
Volume 3
Series Editor: JAY R. GOULD

Baywood Publishing Company, Inc.
Amityville, N.Y. 11701

Copyright © 1988 by the Baywood Publishing Company, Inc. Amityville, New York. All rights reserved. Printed in the United States of America.

Library of Congress Catalog Card Number: 87-27308
ISBN: 0-89503-049-7

Library of Congress Cataloging in Publication Data
Main entry under title:

Word processing for technical writers / edited by Robert Krull.

p. cm.—(Baywood's technical communications series; v. 3)
Includes bibliographical references.
1. Technical writing. 2. Word processing. I. Krull, Robert, 1947- . II. Series.

T11.W67 1987 87-27308
652'.5—dc19 CIP

ACKNOWLEDGMENT

Several people contributed to make this volume possible aside from the editors listed on the title page. Philip Rubens in particular played a large role by writing several section introductions, editing chapters, keeping in contact with authors and the publisher. Without his help, this book could not have been finished.

A few authors submitted manuscripts that we were not able to use. This was not due to their manuscripts' lacking quality and I would like to thank them for their efforts, though I cannot mention them by name.

Finally, I would like to thank Baywood Publishing Company for its continuing support during the arduous process of producing a book such as this. It turned out to be more difficult than any of us imagined. Word processing has changed so rapidly that we had to change manuscripts to keep up. Technical writers' using word processing is different now than three years ago when we first thought a book like this would be useful.

Table of Contents

Introduction
Robert Krull 5

PART I: IMPLEMENTING WORD PROCESSING

1. Writing with a Word Processor: Why and How to Get Started
Tom Brownell 11

2. Word Processing as an Investment in Quality
Charles E. Beck and John A. Stibravy 20

3. Word Processing for the Technical Writer: A Case Study
Will Wheeler 31

PART II: ORGANIZING TO WRITE

4. Strategies for Word Processing in Technical Communication
Charles R. Fenno 47

5. Using Electronic Writing Aids as Editors
Robert Krull 61

6. Beyond Word Processing: Computers in the Composition Process
Frederick M. O'Hara, Jr. 72

PART III: GRAPHICS AND ELECTRONIC PUBLISHING

7. Using a Word Processor for Page Design
Patricia Caernarven-Smith 91

8. Computer Graphics for the Technical Communicator
William L. Benzon 115

9. Text Preparation and Transmission for Word Processing
David T. Orr 138

10. Word Processing and Electronic Publishing
Philip Rubens 146

EPILOGUE

11. Learning to Write with a Word Processor
Annette Bradford 160

Contributors 170

Introduction

ROBERT KRULL

Word processors come with manuals describing functions and how people can access them. They are written for a wide range of users and illustrate how to generate and manipulate text, how to store text in computer files, and how to produce an adequate copy on a printer. These are all valuable skills and technical writers working with a word processor will have to know them; but they are not enough. Technical writers' work differs from that of many word processing users' in the amount of information collected prior to writing, the complexity of writing procedures, and the character of the final product.

Technical writers collect a lot of information prior to producing first drafts. Many word processing manuals demonstrate features with a simple business letter whose original information comprises at most two paragraphs. These manuals give little attention to information collection, organization, and integration.

Most technical writers work with at least one other person. Typically they interact with technical personnel, other writers, editors, and clients. The sample business letter in most word processor manuals is typed rather than composed, alone by one person, seemingly with little concern for the letter's effectiveness. Missing from the manual are discussions of the efficiencies word processing provides for texts written and frequently revised by teams. Finally, while some technical writing products are one page long, many are book length and have to be polished to book quality. This requires graphics, headers, footers, typeface variation, indexes, tables of contents, consistent layout, and electronic typography. The word processing manual's sample business letter contains almost none of these features. To be sure, most manuals describe how to invoke them; but the description stops after keyboard commands and their variations. Little or no attention is given to how to use these features effectively.

These and related issues are the substance of this book. The book is divided into three sections: "Implementing Word Processing," "Organizing to Write," and "Graphics and Electronic Publishing." Each section is preceded by its own introduction, so we will only discuss the section structures here.

The first section, "Implementing Word Processing," offers reasons and procedures for adopting word processing technology, and discusses writer training problems. Many technical writers are hired by companies that provide word processors or may work for ones that change from pencils or typewriters to word processors. Individual writers generally are not consulted about which word processor is chosen. Nevertheless, they may find it helpful to know how their word processor compares to others they have heard about and how difficult they may be to learn. The section also describes basic features of word processing systems most useful to technical writing tasks. Finally, the section shows how writing teams can use word processors to maintain a common style, share text for editing and printing, and evaluate writing quality.

"Organizing to Write," the second section, deals exclusively with writing and editing. It suggests how writers can collect and store information electronically. This information later can be rearranged to form the basis of reports and manuals. Both word processors and other computer programs, such as data base managers, can be used at this stage by technical writers.

The section also discusses how writers can use electronic aids to produce and edit their first drafts. For example, it illustrates how spelling and grammar checkers can help writers detect writing problems. Finally, it demonstrates how to keep track of text files and how to share them among writers using computer networks.

The third section, "Graphics and Electronic Publishing," clearly goes beyond the limits of the business letter. First it shows how writers can make their pages readable, retrievable, and attractive. For example, it suggests how to control highlighting, running heads and feet, margins, headings, and so on. The average word processing user haphazardly accesses these tools. The technical writer must know, not only how to get a word processor to perform these functions, but also how to perform them for maximal reader effect.

Technical writers also need to integrate graphics with their texts. The average writer may paste in graphics. With modern equipment the technical writer can generate simple line drawings that look professionally done. Chapters in this section show how writers can combine their own electronic graphics with word processed text and professionally produced graphics.

Word processing's newest direction is electronic publishing. In its simplest form, electronic publishing involves a writer's adding typesetting codes with a word processor and then sending the text over communication lines to a printer. This saves rekeying text and its attendant errors. Unfortunately, it also introduces incompatibilities among different word processors and typesetters. This section suggests means of minimizing these problems.

In a more elegant form, electronic publishing becomes desktop publishing—text, graphics, and finished copy are all produced at one workstation. When this technology is presented merely as an extension of word processing, the suggestion of increased writer powers is tantalizing. But how well are writers equipped to use desktop publishing? The final chapter in this section addresses this question.

The epilogue describes one writer's learning to use word processing systems. Her experience is similar to that of many writers who have approached electronically-supported writing without technical training. We think that in the future technical writers will become as familiar with the new directions of word processing–text and graphics integration, desktop publishing, and electronic transmission of texts.

We think these chapters cover the current state of technical word processing. We recognize that technological developments will enable writers to do more with less effort in the future, but that, while many writers now do simple word processing, they could be more effective. We hope this book encourages them to be more daring.

Our chapters represent experts' opinions rather than exhaustive studies of word processor usability. It was not our intention to debate issues, but rather, to represent them through diverse, informed viewpoints. The result is some overlap among chapters and some disagreement among authors. We see this as beneficial. Not all technical writers will find the same word processing features indispensable, nor approach writing in the same way. Our book's purpose is to help writers make optimal use of word processing for their purposes, not to brandish a particular writing system.

PART I

Implementing Word Processing

Technical writing can be done with little technological support. Equations, for example, may be more easily produced by hand than with typewriters. However, modern electronic writing and illustrating tools can speed technical writers' tasks and expand their powers. Limited word processors enable writers to spin out legible first drafts; ancillary tools allow writers to generate their own simple graphics. Electronic publishing, word processing's newest extension, gives writers the power to add book quality to their texts. With all these new tools available, one wonders why some writers still work with old methods.

One reason is training. Economical and easy-to-use word processing has been available commercially for at most ten years. It has been available to educational institutions for about five years. Most writers, therefore, were trained to write with pen, pencil, or typewriter. Their use of word processors requires retraining, with its attendant costs.

Another reason not to use word processing is technological limitations. Early word processors were not capable of handling technical writing tasks. They were limited to short documents, offered little typographic control, did not allow sharing text among several writers, and could not integrate text and graphics. These word processors were not used simply because they couldn't do the job required. Modern systems do not have these limitations.

A third reason is fears about how word processing might affect the value of technical writers' skills. Competent writers can, with reason, expect to be at limited capacity while they learn word processing. Writers accustomed to working with pen and paper could expect to lose their writing "feel" when stabbing at keyboards linked to ephemeral dots on a screen. This temporary diminishing of writers' skills could reduce self-esteem as well—after all, trained typists may initially outperform experienced writers.

The chapters in this section were written by people who made that transition and would not go back. They encountered the very real difficulties in using word processors and suggest how to overcome them. They also discovered positive changes in their writing due to the word processor's influence. Most important, they show how word processing features can be optimally used in technical writing.

What is the difference between a word processing system and a word processing program? In Writing With a Word Processor: Why and How to Get Started, Tom Brownell distinguishes between word processing systems consisting of integrated software and hardware and word processing programs consisting of one kind of software that can operate on microcomputers. He also suggests how word processors' text storage capacities affect their suitability for technical writing jobs. Finally, he describes text printing options from preliminary draft quality, through correspondence quality, to book quality type.

Even people who don't use word processors are aware that clean copy is a major advantage over hand-written drafts. Fewer know that word processing can affect writing more profoundly. Charles E. Beck and John A. Stibravy report such findings in their study of Word Processing as an Investment in Quality. Their chapter shows that new users do obtain the clean first drafts they expected. However, it goes deeper to detail changes in writers' outlining prior to writing, their revising text, and their attitudes towards the tool. Beck and Stibravy make clear that the word processing option is an evolutionary integration of new and old skills rather than a simple, one-time step from the old to the new.

In his chapter, Word Processing for the Technical Writer: A Case Study, Will Wheeler describes how word processing fits into the operations of a company. He shows how the word processing system used by a technical writing group differs from that needed by an individual and how it can be used to adhere to the group's style guide. He also suggests training and security guidelines that, while changing how a company operates, also maximizes word processing's advantages while minimizing usage problems.

After completing this section, readers will be aware that others like them have taken advantage of the word processor option. They will be able to discriminate among major equipment classes and how well they fit their writing needs. We hope they also see the advantages of word processing.

CHAPTER 1

Writing with a Word Processor: Why and How to Get Started

TOM BROWNELL

THE MICROPROCESSOR AGE

It's the microprocessor age and computers have entered your office. What you see on the screens of colleagues who are working on computers looks like a typewritten page and you hear the nearly unanimous claim by all addicted computerphiles that they'll never go back to their typewriters or pads and pencils again. Time to take a closer look, you say—maybe it's even time to learn how to write using a word processor yourself.

Before plunging into the word processing mainstream, let's consider what writing on a word processing system will do for you. First of all, you'll experience noticeable strides in productivity, which is, after all, the primary reason for the word processor's popularity. Nearly everyone who uses them, from Isaac Asimov on down, and all the studies say that writers who use computers accomplish more in less time. Why will you write more, or faster, using a word processor?

There are two reasons. First, a word processing program works as a partner with the brain's creative activity. When you write with a pad and pencil you can jot down the ideas that spring into your mind (if you're quick), but at the risk of turning the paper into scrawlings of jibberish. The typewriter, although it has the advantage of rendering your communication readable, stiffly limits creativity. Just think that you'll make an error when typing and your fingers seem programmed to depart from their intended path. Creative writing and plunking letters on paper with a typewriter are the antithesis of one another. Once an idea is typed it's as much as set in concrete. Go back and change how you've said something and you'll probably find that you want to change something else. Pretty soon, the rest of your day will be lost retyping a single page.

Fluid Text

While the word processor shares the typewriter's ability to render information legible, it is wonderfully forgiving of errors and responds to the mind's creative spontaneity. When your fingers stumble or rest too heavily on a key, the computer allows you simply to back up and erase the deviant character. After all, to the word processor letters and words on the screen are only dancing electrons. Because the text you see on the display (commonly called a CRT–for Cathode Ray Tube) is no more permanent than your thoughts (until it is stored on disk or printed on paper) the words and sentences can be altered, moved, shifted, rewritten, and rearranged at the press of a few key strokes. These capabilities, which are available only through a word processor, make writing fluid and allow writers to play with words and their arrangement until the thoughts those words are intended to express ring clear. It's this ability to hold ideas in a fluid state that gives writers who use word processors the creative and productive edge. Because writing with a word processor allows your ideas to form as they do when you mull over the right way to say something in your mind, the outcome is very little unproductive time. Who cares what words go up on the screen when you start, or when you're grasping for the right way to present information? The words aren't set in stone; they can be changed quickly and easily if better words or a clearer way of expressing the concept comes to mind either while you are composing, or later during revision. The absence of mental log jams–wondering how to start or what to say next–stands as the second big reason why those who write with word processors are more productive than those who don't.

Quality Text

Besides being more productive in your writing, and feeling more at ease with writing tasks, another strong attraction of writing on a word processor is the ability to have total control over the quality of the writing that leaves your desk. In offices where secretaries or word processing typists handle the task of transcribing pencil and paper or typewritten drafts to clean copies, writers must correct someone else's mistypings and textual errors then send the marked up pages back for retyping and around and around you go. When you write on a word processor you're the master of your own final copy destiny.

It's one thing to be responsible for your own errors, and another to have the errors of others reflect the quality of your work. With a word processor you will enter your copy, revise, correct, and print that copy. Gone will be the variances resulting from a secretary who couldn't read your writing, understand your instructions, or didn't try. Important to those tight deadlines, turn around time on making revisions or corrections is as fast as you do the work yourself. And should the word processing software in your office also be fitted with a spelling dictionary, the tedious job of ferreting out misspellings and typing errors becomes a breeze. (See Bob Kroll's essay for more on spelling checkers).

Boilerplated Text

For repetitious office correspondence, such as letters to clients or employees that contain a core of standardized information, but must be individually addressed and require other tailoring, the word processor is an ideal tool to crank out the work. With a word processor, just the text that changes needs to be altered; the rest can be left unchanged. So each "new" letter takes no longer than the time required to write the individualized portion. Reports, too, sometimes have repetitious features and here the ability of the word processor to copy and move blocks of text both within a portion of writing and from one document to another comes into play. But before examining the many features of word processing that make writing easier, and more enjoyable, suffice it to say that the decision to use a word processor as an extension of your ability to create written communication simply makes sense. To turn away from the electronic writing wizard is nigh unto relishing the bygone days of the brontosaurus and companion thunder lizards. Using a word processor—or pressing for the availability of a word processor in your office—is the right writing decision. Once made, it's time to turn your attention to features you can expect to find on any professional word processing system and others you may wish to see included.

WORD PROCESSING SYSTEMS

Essentially three types of word processing systems are found in today's offices. The first are dedicated word processors. These are desktop computers capable of text entering and revising tasks, but little else. While most dedicated word processors do their intended jobs well and usually display text sharply and clearly on the CRT, the ability of personal computers to serve a broad range of office uses and the rapidly diminishing cost of multi-function desktop computers has turned most businesses from purchasing single-purpose word processing hardware.

A second approach to office word processing, the use of "dumb" satellite terminals that are linked by a cable or phone line to a back room mini or mainframe computer, is also giving way to the third, and most popular form of current word processing installations.

Today's personal computers (commonly called PCs) have the computing power and information storage capacity found just a few years ago in refrigerator-sized mini-computers. As a result, PCs can be used for a variety of office tasks besides word processing, including accounting and record keeping uses. As writers we shouldn't dismiss accounting and data management programs as tools useful only to financial and managerial personnel. Spreadsheet programs that tabulate information on the basis of quantity and cost can serve writers well in budgeting and keeping track of time spent on projects. Data bases are extremely

useful for organizing information, be it client addresses, reference sources, any details that can benefit from ready access.

Along with the ability to serve as multi-use office machines, personal computers have the added capability of being tied together in a communication network. This arrangement has a number of advantages, such as letting all the writers in one department transmit information to each other directly on computer (thereby eliminating paper mail), setting up a shared information base (for latest product specifications), and can reduce cost by linking multiple terminals to a single printing device.

Two patterns clearly mark the direction of the personal computer industry: expanding computing power and networking. The goal of most computer companies, and the office computing arrangement that most of us will see in the near future, is a personal computer on our desktop or in conveniently situated workstations, with these PCs linked interactively with each other, and perhaps to whatever mainframe or mini-computer hardware our organization may have, and/or to external computer data bases via phone lines.

The word "interactive" expresses a key computer concept. Rather than just being a plastic box occupying space on your desk, the interactive computer responds to you. Early computers were only dully responsive to the information fed them. At one point in the computer's development, information was actually read into the computer by switches that registered the input as raw electronic data. Although a slight improvement, the later keypunch cards housed data in a form discernible only to the computer and branded computers as infamously impersonal machines. Today, using keyboards and video displays, we can see the computer's response to the information we feed it instantly. Tomorrow, interacting with a computer won't be limited to tapping the intelligence of just one machine; the commands we type will scan the memory and stored information of dozens, or even hundreds of computers for research, updating, and scores of other uses. Before long we'll be talking to computers, and most likely they will be talking back.

WORD PROCESSING PROGRAMS

Potentially useful as computers are, their ability to perform a desired action (format and print a letter, for example) relies on programmed instructions called software. Programs for writing with a computer fall into two basic categories: text processing and word processing. Although the terms *text* and *word* processing are often used interchangeably, there are basic design differences that distinguish these two forms of writing software. Text processing was the earliest method of word handling by computer, and many of the primitive examples found in former days on mainframe computers were quite cumbersome. Acting as little more than line editors (writers could make changes on each line as it was typed—an editing capability much like what's available with a memory

typewriter), the early text processing programs had limited ability to format documents and required users to memorize a muddle of arcane structuring commands to turn out a product as simple as a business letter. More modern text processing programs give writers substantial control over the appearance of their documents and can manipulate chunks of text with ease, but most text processors still require coding commands to be inserted into the text for margin widths, spacing between lines, and other formatting requirements.

The second difference between text and word processing software is that text processors usually break writing and printing into separate tasks and sometimes handle the two activities by different programs. As a result, the document as seen on the CRT may bear little resemblance to the printed product. When working with text processing programs, writers may need to use their imagination or rely on their familiarity with the software to visualize the appearance of the finished product.

With word processing, what you see on the screen is exactly (or almost so) what will appear on the printed page. The computer buzzword WYSIWYG (an acronym from What You See Is What You Get) expresses the word processing concept. Because the document on the screen is a literal representation of the paper product, appearance considerations such as whether a numbered list will complement or complicate the information flow can be determined during the composing process. In addition, since a secondary program may not have to be called up to handle printing tasks, start-to-finish writing tends to be faster when using word processing software.

Word Processing Features

Whether the writing software you work with is of the text or word processing variety (in this chapter the term *word processing* will be used generically for both types of writing software) a number of standard conveniences can be expected. When the computer was first adopted as a writing tool, its ability to insert or delete characters, numbers, words, paragraphs, or pages without disrupting the existing text was quickly seen as the microchip's greatest contribution to writing freedom. Just pause for a moment and consider what the ability to insert and delete at will means to your writing—no more crossing out or erasing, no more messy erasure fluid or "snow-tape," and most importantly no more fear of making mistakes. When an error occurs or changes need to be made, the cursor (a moving bar or blinking box) is simply positioned where the alteration belongs and the correct information is typed. Revision is that simple.

While inserting and deleting commonly occur during writing and editing, moving text is a frequent revising function. Working on paper, a writer reorganizes information by circling items that need to be moved and drawing arrows to the new location, or literally cutting a document apart and taping the pieces back together. The computer can likewise rearrange information, but does so in an instant and leaves the reorganized document ready to print.

With most word processing programs, here's how the move function works. To start, the cursor is placed at the beginning of the text that will be moved. Next a function or specialized key is pressed to mark the text and the cursor is respositioned at the end of the selection. Now another key (or key combination) is pressed to designate this block of text for insertion in a new location. The cursor is then moved to where the text is to be placed and another function key (or combination) pressed. Pop! The text appears. Sections of text can be *moved* (in which case the original is erased) or *copied* (the original is retained). Some programs allow writers to move or copy text from one file (or document) to another. Besides reordering information the move/copy function takes the tedium out of repetitious writing tasks, such as letters that share a common information core.

A third feature found in nearly all writing software is the search and replace function. Here, too, the computer swiftly and effortlessly performs what to us humans is droll and tedious. Search/replace is used, for example, when you are proofing a document and find that you have misspelled a name. Will I be sure to catch all occurrences of the wrong spelling, you wonder? Although computers are subject to the errors of humans who use them, once set on a task they are unflaggingly accurate. When writing with a word processing program, all you need to do to correct all occurrences of a misspelling is to activate the *search* function, tell the program what word to look for, then type in the correct word and activate the *replace* function. Virtually as quickly as you can blink your eye, the program will jump through the document, highlight each searched-for word and ask if it's okay to make the replacement, wait for you to respond, then go on. When all occurrences of the word have been located and corrected, you can proceed with confidence that no misspellings remain.

The search function has other uses such as allowing you to jump through a document by referencing key words (headings, for example), and for easy, accurate indexing. By performing dull tasks like these, the word processor releases a writer's spontaneity and opens the door to what makes communication work.

The single disadvantage of writing on a computer–one that is trumpeted more loudly than need be–is the fact that only a small portion of text (typically 24 lines) can be seen at a time.The rest of the document can be viewed by scrolling it through the window of the CRT. Actually, when you think of it, a writer isn't continuously thumbing back through former passages of text when writing on paper, so the twenty-four line window presents little hindrance. When editing, the best approach is to print out a paper copy and either rough-edit the printed version or refer to the printed copy when editing on the screen. To help overcome the limitations of the twenty-four line window, some programs allow you to set a "bookmark" so that when you move to examine other parts of the document you can return to your departure point with a keystroke.

Word processing programs indicate page breaks variously, usually by a symbol set in the left margin to mark the line of text that ends the page. All professional-grade word processing programs will calculate page breaks automatically based

on the standard sixty-six lines per page. Some programs enable you to change the lines per page setting for printing forms or longer sheets of paper. Although automatic pagination is convenient, there are times when the page breaks set by the computer are not suited to the document. For example, it is altogether possible that a heading could appear on the last printed line, with text following on the top of the next page. In most programs, this undesirable separation can be corrected by setting a "hard" (forced) page break that places the heading with the text at the top of the next page.

Along with pagination, all professional-grade word processing software will number the pages and may allow you to set an initial number on which the program will base succeeding page increments. This feature enables you to create long documents as a series of separate files and number the pages consecutively from one file to the next.

Many occasions in business writing call for header or footer information to be placed on successive or alternating pages. The ability to print running heads and place them where you choose is another feature you can expect on any professional quality word processing program. Most programs will ask you to set header or footer information at the beginning of the file (the document you are working on). Apart from the place where you insert the information to be contained in the running heads, you won't see the headers or footers (or page numbers) until the document is printed.

You can expect a word processing program to serve you with a number of other conveniences, including the facility to reassemble a paragraph of text instantaneously, or at the press of a key, after you have disrupted the original by inserting or deleting. Some software even allows you to correct common errors like transpositions, where *their* may have been typed *thier*, by placing the cursor on the first character that is out of sequence and pressing a key to activate the transpose function. Virtually all programs have a repeat character capability, meaning that when you press and hold a given key, its character or symbol will be typed repeatedly until you release that key. This function may sound similar to the repeat action of an electronic typewriter, but there is a difference. A computer works so much faster than a typewriter that the internal electronics will quickly get ahead of the characters you see repeated on the screen and soon you will hear a series of distress beeps telling you that the computer has filled its memory buffer where keyboard commands are stored. The warning also usually indicates that you have inadvertently typed more characters than desired.

OUTPUT TEXT QUALITY

Besides easing and facilitating the mechanics of writing, computer word processing is taking rapid strides toward giving writers control over the actual printing and page layout process as well. As a first step in this direction, most of

today's word processing programs allow writers to select from several print styles, to italicize, make characters bold or oversized, or to miniaturize the characters by compressing their size, and to underline. Helpful as the option to select from a dozen or so printing variations is for indicating book titles, emphasizing headings, and the like, programs that provide these print variations are primitive in comparison to others that also give writers the option to pick the type size, arrange text on the page in a variety of formats that include single or double column, and merge graphics with text.

Personal publishing programs that take documents composed through word processing software and format them to a range of type specifications and layout configurations are also available. The final product is virtually indistinguishable from pages that have been professionally typeset. Most personal publishing software and certain word processing programs also enable writers to incorporate graphs and illustrations created by yet other software and print the visuals as part of the total page design. The ability to merge text and graphics and control page layout is the current frontier of computer word processing—an area where exciting and significant development is occurring. (See Philip Rubens' essay for more on this topic).

LEARNING WORD PROCESSING

Learning to use a computer word processor is far less imposing than the technology would suggest. The popular business-grade word processing programs are designed for easy familiarity and most have quick reference instructions that can be displayed on the CRT when questions arise. The basic activity of nearly all word processing programs parallels that of a typewriter—with some variations. Of course, on a computer typing can occur anywhere on the screen simply by moving the cursor or insertion pointer. But otherwise the process of creating text is much the same whether you are using a typewriter or word processor. You will notice, though, that most computer keyboards have additional keys not found on a standard typewriter keypad. These auxiliary number and special function keys may be used for such helpful actions as jumping the cursor to the start or end of a line, or to the top or bottom of the screen, display instructions on how to operate the software, and a variety of other helpful activities.

Manuals for the more popular word processing programs are well written and referenced from numerous orientations so that you can find the information you need quickly and easily. Although manuals that accompany software really aren't intended to be read cover to cover before you start to use the product (you'll quickly become saturated with new information if you try a cover-to-cover reading approach) their contents should be carefully read and studied as you gain familiarity with the program. To acquaint new users with the software, the better manuals begin with a tutorial that takes you through commonly used functions like moving copy, inserting and deleting, searching and replacing

information, and printing. Working through the tutorial before attempting writing on your own will shorten your learning time. If after finishing the tutorial you refer to the manual only when you are in trouble, you will severely limit your ability to use the word processing software to its fullest capability. Once you have grasped the program's basics, you should read the manual as you would any instructional text and continue this practice until you have mastered the software's full capability. Above all, avoid relying on someone else for instructions. If you let a co-worker "mother" you into the software and aren't weaned quickly onto the manual, chances are you will learn but a small fraction of the software's capability and will stumble along relying on someone else's limited knowledge.

For some, a major obstacle to writing with a word processor—or a typewriter for that matter—is lack of facility with a keyboard. Voice response programs are still in the early development stages, but there may come a time when you will be able to dictate to a computer as to a secretary. In the meantime, some word processing programs use a mouse (a pointer device attached to the computer by a mouse's tail-like cord) to move the cursor, call up files, initiate functions, and generally minimize or give an alternative to the keyboard for all but essential typing activity.

SUMMARY

To sum up, a computer word processor works in harmony with the mind's creative communication process, unlike any writing tool previously devised. It makes writing easier, the time spent writing more productive, and helps break the terrible barrier posed by a blank sheet of paper. As to the equipment one should use—if your office is thinking about purchasing a word processing system, tour the computer stores and take a test write on what's there, ask fellow professionals what's in use at their offices, and read the computer magazines lining the shelves of the local newsstand. Even if you are an absolute computer neophyte, there's no reason to be intimidated by articles on computers that you will find in many newspapers and magazines. Most are well written and don't require a computer background to understand the less technical features that describe and compare equipment and software. As you try out various hardware and software combinations, hold out for the match that "feels right" to you. Word processing systems reflect their designers' style; sample enough of them and you'll find a style that is compatible with your own way of thinking and writing. If the word processing system has already been selected and is waiting for your use, hesitate no longer.

CHAPTER 2

Word Processing as an Investment in Quality

CHARLES E. BECK and JOHN A. STIBRAVY

Learning any new skill takes time, and working with a new piece of equipment usually requires a person to change ingrained patterns and habits. Such time and change represent significant hurdles in learning. People often may hesitate to learn new skills or to experiment with the new equipment, for time "lost" in the learning process means time not available for the basic job. Additionally, time lost for learning may bring conflict over priorities (job vs. new skill), frustration, and increased tension. Even more significant than the time lost, perhaps, is the personal change required in the process. New skills or new equipment means unfamiliar territory, moving from the known and safe to the unknown. Although Catano's research on word processing in the schools indicates that students adapt rather quickly, perhaps we of the "older" generation must overcome the sinister disease "hardening of the categories" as we approach word processing.

Moving from traditional modes of text production to word processing is a significant step—even more significant than the move from paper to a typewriter. Typewriters produced a text that was uniform, neat, and precise; but using typewriters often confined the writer. A typed page, particularly one early in a long document, represented a significant investment: since the typist spent hours putting the text into the original format, changes meant redoing the entire document. The reality of typing time forced writers to forego significant changes to the text. In contrast, the flexibility of word processing frees the author from this limitation. Realizing this freedom, a writer can use word processing as a tool for improved writing quality.

The advantage of word processing can best be understood as an investment in quality. As with any investment, there is an initial cost, for the writer must

spend something before realizing a gain. But the investment pays a dividend, the mastery of a tool that can effectively change not only the appearance or the quantity of output but its very quality. This chapter examines the nature of the time investment and explains ways to use word processing in outlining, drafting, and revising a document. The final part of the chapter summarizes the payoff in terms of attitude change and quality improvement as reported in recent research.

INITIAL INVESTMENT

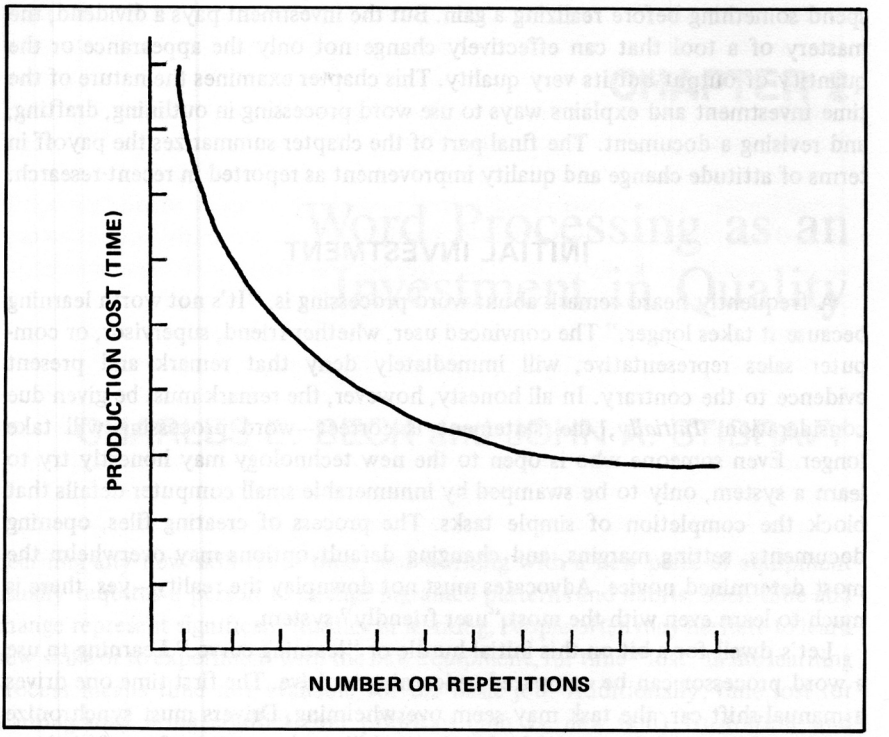
A frequently heard remark about word processing is, "It's not worth learning because it takes longer." The convinced user, whether friend, supervisor, or computer sales representative, will immediately deny that remark and present evidence to the contrary. In all honesty, however, the remark must be given due consideration. *Initially*, the statement is correct—word processing will take longer. Even someone who is open to the new technology may honestly try to learn a system, only to be swamped by innumerable small computer details that block the completion of simple tasks. The process of creating files, opening documents, setting margins, and changing default options may overwhelm the most determined novice. Advocates must not downplay the reality—yes, there is much to learn even with the most "user friendly" system.

Let's dwell for a bit on this initial hurdle or "learning curve." Learning to use a word processor can be compared to learning to drive. The first time one drives a manual-shift car, the task may seem overwhelming. Drivers must synchronize clutch and gas, coordinate the brake if on a hill, and overcome fear of hitting a car stopped in front or behind. The first few attempts bring lurches of the car, stalled motors, panic for driver and passenger, and perhaps a sense of "I'll never get it." After some drives along deserted roads, ventures into low traffic, and many hours behind the wheel, the process becomes automatic. Drivers don't stop to think of every move; rather, they automatically coordinate actions to create smooth turns, accelerations, and pleasant drives. They can even carry on conversations while monitoring traffic and shifting gears. Driving a car follows the classic learning curve. Although initial actions required agonizing time to think through each minor step, practice lessened the time to coordinate movements, resulting in a smooth process with little thought involved. Richard Chase and Nicholas Acquino have characterized such a process as the learning curve theory, which states that

1. The amount of time required to complete a given task or unit of a product will be less each time the task is undertaken;
2. The unit time will decrease at a decreasing rate.

Figure 1 graphically depicts the learning curve.

Many writers experienced this learning curve first hand in their initial attempts to type a report. They spent time learning the system: adjusting margins, planning for footnote space, remembering proper tab settings, and learning position



Source: Richard B. Chase and Nicholas J. Aquilano, *Production and Operations Management: A Life Cycle Approach*, 4th edition, (Homewood, Illinois: Richard D. Irwin, Inc., 1985), p. 349.

Figure 1. Learning curve.

of the letters on the keyboard. Earliest efforts at typing certainly took longer than handwriting the paper. Furthermore, typists feared making mistakes, especially before the age of self-correcting typewriters.

Writers may approach word processing with fears similar to first typists or drivers. Perhaps the word processing program will literally "crash" and they will be out thousands of dollars in lost time, text, and machine repair. Initial apprehension and lack of knowledge magnify the difficulties of learning, making them seem larger than life. Rather than underestimate the hurdle, however, let's acknowledge it—there will be a considerable initial investment in time taken to learn the system. In my own experience, for example, when I was working for an organization that had extensive computer and word-processing capabilities, I tried to learn the system to complete a major report. The magnitude of this feat was so overwhelming that I used traditional typing for the report and ignored learning word processing for an additional two years.

After taking time to learn the basics of word processing, however, the user finds these hurdles less obtrusive. The process then becomes commonplace, not creating any added difficulty. Again using my own experience, I later entered an organization which provided smaller word processors and quickly mastered the system; in a week, I was typing faster than on a conventional typewriter. With the variety of "user friendly" systems now available, the writer still faces a hurdle, but one no more complicated than learning to use a typewriter or to shift a standard transmission. Like the freedom of driving where we want to go, the payoff of this investment brings freedom to produce a document that reflects our training as professionals.

EMPLOYING NEW SYSTEMS

Once an individual has mastered a few basic mechanics of a word processing system, he or she can begin to use the system to greater advantage. Such is the claim of word processing users, and the claim is justified. However, rather than arbitrarily becoming locked into "the way" to use a given system, the writer can assimilate aspects of this technology into his or her own personal approach to writing. Just as people all use pens but have variation in color and type (fountain, ballpoint, felt tip), users of word processors can incorporate the technology into their own personal style of drafting and editing. In that way, the technology becomes the writer's tool rather than the writer's master.

Outlining Stage

For longer reports and projects, authors often use an outline to keep them on track, but the outline may come about in a variety of ways.

Making mental outlines – Some writers feel more comfortable merely using a mental outline or starting to write and letting the ideas come as they will. Analytical thinkers can often work from mental outlines even for long, complicated reports.

Using pencil and paper – The author can prepare a handwritten outline as always, employing the pencil and eraser to great advantage. The familiarity of the yellow pad, a pencil to chew on, the kinesthetic aggressiveness of emphatic underlining and even erasures may enhance the initial creative process. The pages spread out on a desk may give the author a greater visual field for organizing the outline.

Using a word processor – The writer can sit at a screen, type ideas into a new document, rearrange their sequence, add and delete, and come to a final determination of the exact content desired. Additionally, two or three outlines may run concurrently until the author verifies the best approach.

Purists in word processing may insist that the third approach above is the only true use of the word processor. However, there is no single "best means" of organizing ideas for a given report. Based on the technology itself, using a pencil may produce an outline just as quickly as a screen. The chief advantage of word processing is its ease of updating and revising. But with extensive major changes (adding, deleting, inserting) and with multiple iterations, rewriting an outline can become as efficient as salvaging minimal portions of a processed text. The user who selectively employs the processing tool becomes the better writer. Thus, the effective use of word processing during the outlining stage depends on the extent of reworking and the personality of the author.

Initial Drafts

Whether working from a vague idea, mental outline, or written outline, at some point the writer must get the words into a draft. Again, multiple methods exist for drafting.

Screen composition – The word-processing system permits the author to sit at a screen, using an outline or only ideas in the head, and begin to prepare a first draft. Many users feel this stage goes faster with the computer: ideas flow without the author slowing down for details. James Catano calls this concept the "fluid text" and maintains that it differs significantly from traditional text preparation in which ideas all too quickly become static or fixed. Another analyst of word processing, Donald Case, would view this writing-flow approach as one of the dominant cognitive tendencies of the "wholist" vs. the "serialist." The wholist may create a draft to get the scope of the project, then revises heavily; the serialist, on the other hand, outlines in detail, then crafts each paragraph as he builds the document. Thus, a writer's way of using the screen on the first draft possibly reflects a cognitive stance toward writing itself rather than a superficial like/dislike of computers.

Those who adapt to the computer tool at this stage find a sense of liberation, increased speed, and a greater sense of control over the writing process. Among professional writers on a university faculty, Case finds that soon after they learned word processing, 60 percent preferred to compose entirely at the screen.

Handwritten – Many authors prefer to draft on paper rather than on a word processor. For the most part, these persons are comfortable with that method because of an ingrained habit. They approach the writing task with a mind-set or collection of images/feelings that include a piece of paper or a yellow legal pad for productivity. To remove this aid to creativity for such a person could prove disastrous. Butler's research in business-writing classes and Thiesmeyer's with composition courses find that their students prefer handwritten drafts to screen composition. Perhaps the familiarity of this first step eases the transition to word processing for these students. Additionally, Keiffer and Smith's experiment in students' conceptual learning with word processors required the use of a

handwritten draft. Unfortunately, they do not explain the rationale for this requirement; perhaps availability of word processors prevented a more widespread usage.

The word-processing "fanatic" might bemoan the use of a handwritten draft, but perhaps needlessly. Word-processing technology permits a rapid change of text and major revisions. But the greater the extent of revisions, the less time-efficient word processing becomes. If most sentences are changed from 50 to 75 percent, the time spent on numerous commands to add and delete text could match the amount spent retyping the document completely. In such instances, the computer fanatic is hard pressed to justify the advantage of word processing over regular typing.

There need not be, however, any such justification. Either method gets the author into the document. Although current studies have not examined the time spent on drafting by hand vs. drafting on a word processor, I believe that the word processor may have a slight advantage, but not more than slight. The person who drafts by hand and transfers the text into the word processor has just used an alternative means of drafting a text. Additionally, entering the text usually involves so many changes that the product entered on the screen is not just the first typed copy but the second draft. Rather than approach the word-processed draft from an "all or nothing" perspective, the user should acknowledge that either approach will get the author into the text.

According to Case, some evidence suggests that the longer an author uses a word processor, the more initial drafting is done on the machine itself. In time, the writer becomes more comfortable with the word processor, assimilating the machine into his or her approach to writing. Technology thus "changes" the author, as mentioned above. Even though many experience this effect, it is by no means universal. Nor should it be. Adapting technology does not imply that all users experience identical benefits or apply the technology to all stages of the writing process. Essentially an author must integrate the use of a new tool into his or her own writing style; the computer then aids the writer rather than controls the creation process.

Revisions

At this point, all users have a text on the word processor and can use the computer for multiple revisions ranging from minor word changes to major shifts in text. However, as with the other stages of the writing process, there are various methods for using the word processor to revise a text.

On the screen – Some authors do a complete editing from the screen itself, by constantly changing of words or sentences either while drafting or after the document is complete in the file. The limited research on the topic indicates that much "microediting" occurs on the screen; Case and Meeker even believe that it fosters endless tinkering.

Depending on the complexity of the word-processing system, however, the author may have new techniques available. For example, the writer may create temporary changes to compare alternatives, rather than simply change the words for a new text. Some systems permit split screen editing, so that users can keep the original text and look at alternatives. In other systems, users work on a temporary text until they update the final copy, permitting many variations prior to making a final choice. Ronald Sudol stresses the principle of addition to increase effectiveness of this mode: add to a text first, then delete. According to Sudol, writers should never delete anything without first generating alternatives and then juding them in context.

In a limited research study, Harris indicates that perhaps with a screen-only approach the author does not do as much "macro-editing" of large sections of the paper. Writers tinker with word choice, but seldom rearrange major segments of a long text. The findings of Harris, however, are rather tenuous at this point; among other issues, if writers outline accurately, perhaps they do not need major shifts in the logic of the original outline.

From a printed draft – Because of the ability simply to reprint fresh copies of a text, many authors print a copy to "see" the entire text at one time in front of them. After making pen and ink changes on the draft, they type the changes into the file and reprint a fresh copy. Harris finds that authors who revise a text extensively prefer this feature of the system. The word processor keeps the text clean, less messy. Rather than becoming lost in the flow of changes to a draft, the author can update the copy, and work with a new text.

The "paper-draft" user perhaps sees the word processor as a sophisticated typewriter; however, writers become more willing to change the text, knowing that the retyping is no longer a monumental task. Beck and Stibravy found that extended use of word processing was likely to change a text, producing new clean drafts for further review. The clean copies keep the author from getting confused on major adjustments to the text. In this way, word-processing technology contributes to improved quality of writing. Again, the purist may insist on using screen editing exclusively, but such a limitation is unnecessary. Using the reprint capability may require incremental costs in ribbon and paper, but it gives authors an alternative means of using the technology to enhance the creative process.

Mixed style – The final variation is a mixture of the two approaches. In this mode, authors may read through the document on a screen, making minor changes to the text, but reprinting after they make major changes.

Ultimately, authors use the technology as a tool, adapting it to their own comfort as writers. Gingrich's study of professional writers and Butler's of students indicate that soon after learning a system, users tend to adapt system capabilities to their own needs. Rather than be forced into a single use of word processing, users tend to use technology as tools incorporated into the writing process.

CHANGES IN ATTITUDE

Part of the adaptation to word-processing technology seems to represent a change in attitude not only toward the use of computers but toward the writing process itself. Most published studies of word processing indicate that users experience a positive change in attitude toward writing. For example, Meeker's study of writing behavior indicates that word processing encourages revision and reorganization of ideas, eliminates the anxiety of the blank page, provides a sense of mastery over the document, provides a sense of freedom from physical restraints of handwriting or typing, and brings a potential for change in writing style. In a cautious summary of word processing research, Cross and Currey state the following:

> Efforts to use word processing will be continued because it is a promising concept that fits very well with the current emphasis on *process* in writing, people like it, no major failures have occurred and some positive changes in writing behavior have been observed.

At worst, according to these observations, word processing has not revealed any negative attributes which hinder writing; at best, it provides a tool that users like and often encourages their writing.

Among specific groups of users, only minor differences appear between attitudes expressed by beginning vs. more experienced writers. Among freshmen who used the UNIX ™ editing system, Kiefer and Smith found that participants "overwhelmingly agreed that using the computer was enjoyable, easy, and not frustrating. Students also agreed that they would look forward to another composition course using computer assistance and that they were learning more about style and diction." In a separate study by Harris, students found computers faster, easier, and neater; furthermore, students basically enjoyed the writing. Harris also found, however, that overall major revisions were reduced by word processing—a finding that so far is not substantiated elsewhere. According to Rodrigues, anxiety lessened as the semester progressed: journal entries later in the course showed a significant reduction in anxiety about writing compared with earlier entries. Most significantly, however, computer use encouraged independence: "In the last few weeks of the semester, students rarely asked for help with word processing or with writing. They knew what they wanted to do, and they had the confidence to attempt it by themselves."

Among graduate students in business and engineering, (most of whom were experienced managers and report writers), Beck and Stibravy found that users saw computers as an aid to composition, a means of saving time, and a more "fun" approach to writing. Using computers encouraged students to make changes to their own texts. Butler's study of business students who had not previously used word processors revealed that 96 percent changed their perceptions of the writing process, resulting in more positive attitudes toward writing. In a similar manner, Case's study of faculty users revealed an increased motivation

to write, for writing became "more fun" and "less frustrating." Overall, these findings support overall conclusions that word processing affects user attitudes, with the vast majority viewing writing in a more positive manner.

INVESTMENT IN QUALITY

The great difficulty in research in word processing is concretely evaluating the effect on quality. The controls needed for a strict quantitative study are difficult (some would say impossible) to design. Consequently, those who examine the issue normally resort to user preceptions of quality change.

In existing studies, improved writing quality is discussed more frequently among advanced writers and professionals than among new users. According to Case, university faculty members believe that word processing has improved the quality of their writing. Similarly, Catano concludes a study of two creative writers with a rationale for quality improvement: word processing blends the collection and forming of research materials, enhancing the creative process. Among professional writers using the UNIX[tm] editing system, Gingrich reports that users became more aware of principles of good writing and learned how to change their style for particular audiences. In addition, users determined which aspects of the computer to incorporate into their own writing style, tending to exploit those programs that took the tedium out of writing.

One long-term study of users at the start of a master's program and thirteen months later indicated a major shift in attitudes toward writing quality. At the beginning of graduate study, users were drawn to word processors as "time-savers" or "fun way to write." At the end of a year (without institutional instruction in word processing), these same users identified numerous ways in which word processing improved the quality of their writing in such areas as organization, spelling, grammar, coherence, and appearance. Rather significantly, these users indicate that word processing takes more time than regular typing: this time factor reflects their tendency to make more revisions, significantly changing the text. These writers did not merely assert an improvement in quality; they also listed specific quality improvements and indicated increased time spent in rereading and revising papers as well.

Much of the study of word processing has come from English teachers, persons who may be concerned about their own role in the writing process. In concluding her positive study and endorsement of word processing, Rodrigues makes the following comment:

> Would these students have done equally well if I had sent them off to learn word processing and then returned to business as usual in the class? I think not. The mere availability of computers and the knowledge of word processing alone won't make an automatic difference in students' attitudes toward writing. Surely the English teacher counts, too.

Such a comment, however, does not match Rodrigues's own comments recorded earlier: her students developed confidence and, as the semester progressed, sought less help both in writing and in word processing. Beck and Stibravy's study of graduate students indicated that experienced writers, having learned basic word-processing skills, could improve writing quality without additional institutional involvement. Similarly, Gingrich's study of professional writers noted that soon after learning a system, writers became selective in using its features. Studies at all levels of writing experience thus indicate how writers shift their attitudes toward writing as they incorporate technology as a tool.

The field of word processing is new and we have much to learn, but our current research, however limited, is encouraging. Similar evidence comes from students in freshman composition, in business writing, and in graduate engineering or management; it also comes from college faculty across academic disciplines, and from professional tech writers with diverse levels of experience. The evidence points to a process that can apply to all potential users of word processing technology:

1. initial familiarity with systems brings a tendency to increase use of the system;
2. use of the system produces a changed attitude toward writing, with users becoming more willing to refine and correct a text;
3. users rather quickly adapt the technology into their own preferred writing style, making the machine a tool for their own use rather than a controlling master;
4. time and consistent use produces a gradual increase in use of word processors; and
5. users recognize quality improvements in their work and become more articulate at explaining the nature of the quality changes.

Learning word processing takes time, as does learning any new skill. Rather than a time loss, however, the majority of users develop an increased sense of mastery over the writing process. The professional writer has the added advantage of experimenting with a variety of writing approaches (wording, sequence, and layout of the final text) to best meet the needs of supervisor or the intended client. The initial time invested expands the writer's abilities, giving more options to consider in producing a final text. Rather than fearing the computer control, users experience their own sense of mastery over their ability to develop ideas and use tools to refine them.

With experience that permits the user to employ even a part of word-processing capability, the combination of machine and user can improve writing. Rather than a time sink, word processing can become an investment in improved writing quality.

BIBLIOGRAPHY

Arms, Valarie M. The Computer Kids and Composition, paper presented at the *Annual Meeting of the Conference on College Composition and Communication*, San Francisco, California, March 18-20, 1982. ERIC #ED 217 489.

Beck, Charles E. and John A. Stibravy. User Perceptions of Improving Writing Quality Through Extended Use of Word Processors, in *Proceedings: 1985 Air Force Conference on Technology in Training and Education*, American Defense Preparedness Association, Washington, D.C., pp. III-59-III-70, 1985.

Beck, Charles E. and John A. Stibravy. The Effect of Word Processors on Writing Quality, *Technical Communication*, *33*:2, pp. 84-87, 1986.

Butler, Marilyn. One Approach to Word Processing in Business Writing Classes, *Proceedings of the American Business Communication Association Midwest Regional Conference*, Ames, Iowa, pp. 193-200, 1984. ERIC #ED 246 474.

Case, Donald. Processing Professional Words: Personal Computers and the Writing Habits of University Professors, *College Composition and Communication*, *36*, pp. 317-322, 1985.

Catano, James V. Computer-Based Writing: Navigating the Fluid Text, *College Composition and Communication*, *36*, pp. 309-316, 1985.

Chase, Richard B. and Nicholas J. Aquilano. *Production and Operations Management: A Life Cycle Approach*, Richard D. Irwin, Inc., Homewood, Illinois, 1985.

Cross, John A. and Bob J. Curey. The Effect of Word Processing on Writing, paper presented at the *Mid-Year Meeting of the American Society for Information Science*, Bloomington, Indiana, May 1984. ERIC #ED 247 921.

Gingrich, P. S. The Unix™ Writer's Workbench Software: Results of a Field Study, *The Bell System Technical Journal*, *62*, pp. 1909-1921, 1983.

Harris, Jeanette. Student Writers and Word Processing: A Preliminary Evaluation, *College Composition and Communication*, *36*, pp. 323-330, 1985.

Kiefer, Kathleen E. and Charles R. Smith. Textual Analysis with Computers: Tests of Bell Laboratories' Computer Software, *Research in the Teaching of English*, *17*, pp. 201-214, 1983.

Meeker, Michael W. Word Processing and Writing Behavior, paper presented at the *Annual Meeting of the Minnesota Council of Teachers of English*, Mankato, Minnesota, 1984. ERIC #ED 250 686.

Rodrigues, Dawn. Computers and Basic Writers, *College Composition and Communication*, *36*, pp. 336-339, 1985.

Steele, Louise W. The Least You Should Know About Office Machines (A Talk to Students), *The ABCA Bulletin*, pp. 11-17, 1984.

Stibravy, John A. and Charles E. Beck. The Question of Quality: New User Attitudes Toward Word Processors, *Proceedings: 31st International Technical Communication Conference*, Washington, D.C., pp. RET31-RET33, 1984. ERIC #ED 246 661.

Sudol, Ronald A. Applied Word Processing: Notes on Authority, Responsibility, and Revision in a Workshop Model, *College Composition and Communication*, *36*, pp. 331-335, 1985.

Thiesmeyer, John. Teaching with Text Checkers, paper presented at the *Spring Conference of the Delaware Valley Writing Council and Villanova University*, Villanova, Pennsylvania, 1984.

CHAPTER 3

Word Processing for the Technical Writer: A Case Study

WILL WHEELER

There is no question that word processors have released writers from the drudgery of repeated revisions and recopying. The result has been improved productivity. Proofing documents has become easier since it's possible to ensure that a typist hasn't slipped new errors into a document while copying it.

Word processors also make it easier to revise existing documents. An example is changing the name of a product; let's say Jetwrite, to Turbowriter. A typewritten manuscript would have to be carefully edited and retyped, at considerable expense, to accommodate this change. With a word processor, a search and replace command is all that's required. Time required? With the right word processor it can be a matter of minutes. As a result, it becomes easy to polish documents, even when working with a tight deadline.

Reusing existing material also becomes much easier with a word processor. Major chunks of one written work can be integrated with another. Boilerplate sections, such as the description of a writer's company, do not have to be retyped each time. They can be merged electronically. Finally, large-scale production work such as typesetting is easier since it has become possible to pass files directly from a word processor to a typesetting machine.

Most technical writers work directly on a word processor or text editing system and produce documents entirely without the assistance of clerical staff. This allows writers to make necessary formatting decisions on the spot. Technical documents rely heavily on special formats to emphasize points or make them more retrievable. Some of these formatting decisions include: using a box or indent for notes, italics for special words, starting a new page, and drafting a chart. Most word processors make such layout features simple to control; sophisticated word processors allow graphics to be included in the text.

This chapter offers an overview of issues that one corporation considered in implementing a word-processing system. These issues include selecting a word processor, word processor procedures, document production, and computer operations beyond word processing. At each step, a general implementation issue is posited and the corporation's reaction to it is offered. In addition, each reaction's relative advantages and disadvantages are assessed for its impact on individual writers. This case study is not intended as a model for other companies. Instead, it is intended to provide cautionary advice for anyone implementing word processing in a corporation.

The company employs 100 people, about average for computer software companies. The publications department has ten writers who produce manuals and marketing materials. Other personnel include a graphic artist and a manager. Most writing is connected either to new product development or periodic revisions of existing manuals. Companies varying in size and line of work will implement word processing technology differently from this company.

SELECTING A WORD PROCESSOR

Decisions about selecting a word processor usually represent a concensus of writers' opinions, corporate interests, technology available, and the persuasive powers of sales people. Direct and indirect tradeoffs are made at all stages in the process. For instance, the writers' primary concern might be functions that support typical writing tasks. The corporation, by contrast, might base its decision on cost, cross-site compatibility, and speed of producing finished text. These perspectives may be at odds and, in fact, may be irresolvable.

Companies and individual technical writers have different needs. Since companies specify the word processor to be used, writers who work for them must reconcile their preferences with the company's.

Corporate Word Processing Requirements

A company generally has the following considerations in mind:

- Cost (software and hardware),
- Long term usability,
- Document file integrity,
- Versatility.

Cost is, of course, a major consideration. A company must take into account the full cost of software, hardware, supplies, servicing, and training. Companies may save money by continuing to use existing office word-processing systems rather than upgrading to the newest ones available. As a result, writers find themselves working around limitations in the system they are required to use.

Cost is the main reason this company chose its word processor. Since the company developed the word processor for sale to business users, programming

costs were covered by product sales and the only additional direct costs incurred were for duplicating program diskettes and manuals. Indirect costs such as software support, maintenance, and revision were placed into the corporate budget as part of continuing product development. Companies which develop their own word processor, but do not sell it to others, would have to cover these costs themselves. Most companies probably would find that strategy prohibitively expensive.

Unfortunately for our company's writers, its word processor was designed for secretaries rather than writers. It optimized typing final drafts of short documents such as letters instead of revision and maintenance of long technical documents. How the word processor was designed for short, simple rather than long, complex documents is described in detail in a separate section.

Long-term usability of word processing software and the text files they produce may be more important than the hardware they work on. For example, many companies plan to replace computer hardware every three years; but old text files must still be accessible. Our sample company, for instance, needs a year to produce a manual. In addition, writers must revise manuals to match product updates. To protect a company's investment, word processing files holding the manual must be usable over the whole period. This company moved files among several hardware systems while continuing to work with the same word processor.

Hardware migration can be transparent for writers, as it sometimes was for our sample company. A more legible display screen should be welcome rather than problematic. But new hardware may also confuse writers. New keyboards may place keys in different positions, leading writers to make errors such as deleting a text section when they planned to insert it. Ideally, hardware changes should be made with an eye to minimizing writers relearning to use their word processor.

Document file integrity is another factor in choosing a word processor. By document file integrity I mean that access to valuable word processing files can be maintained in spite of minor hardware and software failures. Utility programs are available for some word processing programs to resolve technical problems such as bad disks and partly erased files. Special purpose disk recovery programs, operating separately from word processors, can be used to recover from the same errors if the word processor text files have a structure understandable to users. Some word processors store text files in ASCII (American Standard Code for Information Interchange) code. A moderately well trained user can decipher these text files with a utility program by using an ASCII code table. Some word processors store document files with their own special code. Unless a user can obtain detailed information about these files, recovering document files after hardware or software failures may be very difficult.

Our sample company's programmers know the word-processing program's internal structure because they produced it. They have been able to write their own error recovery utilities and can rescue at least part of the most damaged document files.

Versatility is a third consideration in selecting a word processor. The word processor should be able to handle a company's full range of document formats and output media. For instance, in addition to paper output our sample company's word processor has been used to produce special purpose texts such as online reference material. The word processor suits this task because of its extensive highlighting and formatting capabilities.

Versatile formatting capabilities provide numerous word processing commands but can require a long learning period for writers. A word processor that addresses an individual writer's limited tasks will generally be easier to learn to use. Thus versatility is an ambiguous characteristic. On the one hand a word processor should provide the greatest range of possible functions to meet the needs of a particular text production situation; on the other hand it should not have so many functions that learning their use prevents productive work. Versatile word processors should allow writers to implement basic, typical functions quickly and should also provide easy instruction for more sophisticated functions.

Personal Word Processing Requirements

Individual writers have different reasons than companies for preferring one word processor over another. Writers look for:

- Ease of Use
- Productivity
- Creativity

Ease of Use and *Productivity* are closely related. It's a safe assumption that if a word processor is easy to use, it also raises writer productivity. However, individual writers think different products are easy to use. As a result, in any given group of writers, you'll often find a variety of personal preferences. For example, Wordstar, a microcomputer word processor[1], has attracted a large number of writers. Its loyal users claim that its command sequences are easy to follow and remember. Other writers find its commands difficult to learn. Blanket recommendations about the ease of use of this or other word processors, it seems, cannot be made.

Creativity is most important to writers in a personal sense because it makes their job enjoyable. Word processors can help or hinder a writer's creativity. Generally, the more flexible and faster the word processor, the more it can help a writing project; but other features may appeal to writers with particular skills. Writers who revise documents extensively may value a particular word processor for its powerful editing commands, while writers with a talent for layout may value another word processor's precise control over page format.

[1] Wordstar is a trademark of MicroPro International.

Basic word processors are unlikely to provide the range of functions demanded by creative writers. Complex, versatile word processors may have that range, but are likely to require significant learning. Individual writers should select the simplest word processor offering the functions they most want. The easiest way to make such a choice is to review typical writing tasks and to list the word processor functions necessary to support them. One should include document handling capacities and ease of sharing document files with other writers. Once the list is complete, writers can use it to compare word processors to each other.

Companies usually ignore creativity since it is very difficult to assess and is less tangible than cost. For example, the sample company's word processor hindered writers from moving large blocks of text within documents and so constrained their ability to polish manuscripts. Investment in a more elegant word processor represented too high a cost for the improvement in writers' working environment it would yield.

If writers work alone, they can choose whichever word processor they like. But, as I have already mentioned, a publications department cannot allow everyone to make a personal choice. A single word processor needs to be selected to minimize costs and provide document file compatibility. For publications departments, writer productivity and creativity and word processor ease of use may not be as important as cost and the ease of managing writing groups. The result is continuing tension between individual writer needs and corporate concerns.

The Sample Company's Word Processing

When writers first began using personal computers at this company, two floppy diskette drives and 128K of internal memory were standard. The decreasing cost of hardware has made better computers available. Now it is common to have a hard disk system, with greater storage capacity than diskettes offer, and 512K of internal memory. Writers benefit from faster access to their programs and data, and can utilize a wide variety of programs simultaneously. This configuration also allows them to operate the word processing system and the software products they are documenting.

Word processors can be *page-oriented* or *galley-oriented*. The word processor in use at this company is a page-oriented word processor. This means that all writing is done on an electronic sheet of paper, normally sixty lines to a page. Each page is treated as a separate unit and saved separately on disk. Each time a writer wants to go from one page to another, the computer must store the old page on disk and read the new one from disk. For documents beyond thirty pages, page-oriented word processors become unwieldy and difficult to use during editing. Since some of the documents produced at our sample company are well over a hundred pages long, this presents operational problems for our writers.

When producing an initial draft, the page-orientation isn't much of a problem since most of the writer's attention is confined to individual pages. The page

orientation is also easier for new users to understand because the screen resembles what's finally printed.

Page-orientation is not as important for skilled writers, since they work on long documents. When these writers work on revisions, they often have to jump from one page to another. For example, moving a paragraph from the front to the end of a thirty-page document would require sequentially reading and writing these pages to and from a disk drive. This is mildly annoying with a fast hard disk; it can be painful with floppy disks.

By contrast, galley-oriented word processors use a continuous file, all of which is loaded into computer memory at one time. During editing one can quickly go through an entire file when making edits like moving a paragraph. Before printing, the file goes through a formatting process which breaks this file into separate pages. The limitation of a galley-oriented system is that documents exceeding the computer's memory capacity must be broken into smaller computer files that are linked together for printing.

Galley-oriented word processors have other limitations, especially in the displaying of a document on the computer screen. Many personal computers do not have the capability to correctly display the typefaces and formats that can be produced by printers. To do so, a more sophisticated method of screen management, such as bit-mapped screen is required. Other word processors, using cryptic control codes to handle document layout, require experience to understand and make visualizing the final product difficult. For example, with the recent development of low-cost laser printers, writers have a wide variety of typefaces available. However, none of these typefaces can actually be displayed on non-bit-mapped screens.

In general, professional writers need advanced word processor features and find that functionally simple systems present serious limitations. The time required to learn a more advanced word processor, even a large-scale text processor system, can pay large dividends by satisfying a writer's needs over a longer period of time. The word processor developed and implemented by the sample company, being designed for clerical use rather than for the companies documentation groups, gives an example of such limitations. Its slow, page-oriented file access produced continuing difficulty for writers during editing.

WORD PROCESSOR PROCEDURES: TRAINING, GROUP WRITING, USING STYLE GUIDES, AND SECURITY

Training

At the sample company, learning how to use a word processor is usually the new writers' main assignment during the training period. It takes several days to master the tutorials, but these only serve as an introduction. A good technical

writer must be eager to learn and experiment. This includes tinkering with an important tool of the trade, the word processor. Adjusting to writing on a word processor can be a difficult process, especially when trying to master its mechanics at the same time. It's necessary to change many habits and patterns developed over years of using a typewriter or another word processor.

New writers should set aside several days just for learning. Writers should read the manuals and ask questions of experienced users. They should also take time to become familiar with the keyboard and other supporting equipment. Writing should initially involve playing with the word processor. After producing a sentence, writers should then try different variations of it. They should do likewise with the overall organization of a short document. Finally they should create an outline, write a text from the outline, and then move text sections into the several orders. By exploring these paths, new writers will learn word processor functions that help them produce quality documents quickly.

After word processor familiarization, new writers can take on a small writing assignment of several weeks to a month duration. Upon successful completion of this assignment, new writers can be considered trained. Empirical studies have shown that people can use 20 to 30 percent of a word processor's features after sixteen to twenty hours of learning. This level of skill would be adequate for letters and short documents with undemanding formats. However, experience at the sample company has shown that it takes six months to a year to completely master word processors, especially when technical documents contain multiple column layouts, headers, footers, illustrations, and so on.

New writers then can be given a major assignment taking several months to a year to complete. In computer software organizations like the sample company, writers are often hired because a new product is being developed. Since experienced writers are often already engaged in a project, new writers may soon find themselves working on important projects. Yet, at this stage the new writers' orientation to the company and its methods can still be ill-defined. When they start documenting a new software product, writers often need to learn how to use it without much information either about the product or about what the company expects them to write. This type of training can be useful as an introduction to software documentation even though it is extremely informal. Thus, the experience of learning a word processor can be seen as an introduction to corporate style and culture. The ways the company permits writers to learn about word processors also provides an introduction to what and how a company wants writers to work.

Group Writing

In many companies, writers work in groups. Technical writers at this company commonly work on their own; but they must still consider their work as part of a group process. A document's purpose is commonly defined, not by a writer, but by corporate objectives such as the release of a new product.

At the sample company, a writer commonly starts a project with something written by someone else—a set of technical specifications or programmer's notes. As the project progresses, the writer gathers additional material from company sources such as marketing and customer support. These sources can often provide documents that could become part of a manual. Marketing may be responsible for a brochure that can be adapted into the manual's introduction. Customer support may provide a list of common problems. Based on this source material, the writers prepare their own manual. Drafts of the manual are sent out to potential users and company editors for review and comment, and the writer integrates these comments into the revised draft.

Writing teams go through the same information gathering, drafting and editing process. The team approach is necessary for large documents and when projects must be completed within a short period. Experience at our model company has shown that extensive planning and review are necessary to make writing groups work smoothly. A good outline ensures that writers complete their parts of the task, writing neither too much nor too little. A style guide ensures that different writers' sections have the same format and complementary content.

Word processors aid this process. They can be used to generate the outline and style guide and support the revisions for cohesiveness. Since style guides are a complex topic, I will devote the next section to them.

Using Style Guides

Style guides have traditionally been paper documents that catalog prescriptive rules for producing text in a publishing group. In general, they attempt to specify all of the elements and strategies that can be used by writers, editors, graphic artists, and production personnel. The intent of these is to make information content and format consistent across products and among writing groups. Style guides mandate writing style, grammatical conventions, illustration techniques, text organization, layout, presentation, and other concerns. The individual writer, working alone, can create a document style and adhere to it. However, when more than one writer is involved, a style guide becomes essential for ensuring that documents retain coherent content and appearance.

Style guide specification of content, for instance, is so important to some large companies that they spend a great deal of energy producing specialized dictionaries. By comparison, some smaller firms, whose writers are in frequent contact, update terminology on a piecemeal basis. In either case, trademarks need to be treated consistently; and common terms such as diskette, drive, and screen should be used with only one meaning. Acronyms should be distinguished typographically, for example, by setting them in all caps. The benefit will be reduced reader confusion.

Word-processing programs by themselves can do little to implement stylistic conventions. Editors are still needed to read through documents so that

inconsistencies can be identified. Once this manual task is completed, word processors help editors to make changes more easily. In doing so, editors can use auxiliary computer programs such as spelling and grammar checkers to identify some language problems. Their use is described in a later section and in considerably more detail in the chapter by Robert Krull.

Style guides also should describe document layout when writers are responsible for final "camera-ready" copy. Margin sizes, type sizes and styles, and placement of page numbers all need to be specified, as do special formats for footnotes, graphics, and references. Word processors can help implement this aspect of style guides in two ways.

First, a style guide's layout specification can take the form of a word-processing sample document. The text sample can show how writers can reproduce the format exactly. For instance, it can demonstrate how margins are set and to what dimensions, and how and what typestyle is specified for a specific document. When there are several ways for a word processor to have the same effect on the printed page, the sample document can show which method is preferred. For example, page numbers can be typed individually at the top of each page or specified as part of a word processor's header or footer command. The sample document can indicate which method a writer is expected to select.

Sophisticated word processors offer another way of determining that document layout will match company style guides, and without placing as great a burden on writers. These newer word processors allow a company to set document layout characteristics by defining the word processors' "style sheet." Writers just identify the type of text they are producing; the word processor uses the style sheet to set the text section's margins, typestyle, pagination, and so on. The writer does not have to be concerned with layout at all.

Writing a good style guide requires cooperative effort in advance of writing a document. It is possible to bring a document into conformance with a style after it is written, but only with considerably greater effort than had it been written to a style initially. Word processors can be used to implement the style, but people need to design and agree to use it.

Style guides are not an integral part of word processors but they do give an indication of the expected interaction between writers and documents, and of the documents' final shape. As such they can help determine what functions are needed in a word processor to be used by the writers. For instance, if a style guide requires an index, then an automatic indexing feature would be necessary. Similarly, if the style guide calls for a specific dictionary as a standard corporate reference tool, then any word processor spelling checker should either be based on that dictionary or be amenable to being modified.

Style sheets, in contrast, are a recent addition to word processor functions that provide less writer and more corporation control in the application of style guides. Once a style sheet is configured, it controls any or all layout features of a document; the writers apply their time to writing. Obviously, deciding to use

style sheets, a decision probably made at the corporate level, means that a word processor with this capability must be implemented. The implementation process involves software acquisition, determining a corporate document style, designing the style sheets to reproduce the corporate style, and training writers how to use the style sheet feature.

The word processor developed by our sample company did not have style sheet capability. If it had, writers probably would have spent less time formatting and reformatting text in an effort to duplicate the mandated company style.

Security

Any company with a large publications department has a major investment in the security of its documents, including word processing files and master copies. One morning, workers at our model company arrived at work to find the smell of smoke in the air. The building next door had burned down. Although there had been no danger of the fire's spreading, this was a reminder of the need for a good security program.

There are several levels of security. The ultimate catastrophe would be a fire that completely demolishes the building. A more frequent, minor occurrence is a diskette that is lost or damaged. Between these extremes are security for military or highly sensitive industrial projects and development of competitive new products. As a matter of course, staff are required to sign non-disclosure agreements which bind them to confidentiality. Any project which hasn't been publicly announced is considered secret and is governed by this agreement.

Finally, there's the problem of storing word processing files. Diskettes containing word processing files can be stored quite easily, but they need to be well organized to allow access to documents. Stored files also need to be kept current with revisions to the text. If the right documents can be found quickly, it makes revisions and updates much easier. The files are quite useless if their accuracy is in question.

To meet these security needs, the publications department of our model company maintains a library of important materials. The library stores copies of the documentation in diskette form, as well as the actual manuals. Diskette copies of documentation need to be protected from exactly the same hazards as software programs—fire, flood, and bad organization.

Whenever a manual is finished, a copy of the word-processing files is submitted to the librarian. The librarian ensures that all the required information is submitted, records its files, and boxes its diskettes. If this is an update to existing files, the librarian carefully makes the replacement, ensuring that the right diskette is changed. Periodically, the librarian makes copies of all new files in the library. These copies are stored "off-site" in a secure area.

If the building burned down, these off-site copies could be used to reproduce all manuals. Naturally, work in progress would be interrupted. To guard against

this, writers are encouraged to submit files to the library even before completion of a project. Witers also are required to make backup copies of their diskettes, usually on a daily basis. This is especially important for new writers, who have a greater tendency to destroy files inadvertently.

DOCUMENT PRODUCTION

Once a document is written, it needs to be printed out or typeset. Actually, the production process starts even before the document is written. At the conceptual stage, it's advisable to determine the form of the finished document. It's best to have a definite plan before starting out. This is because different production methods require different word processing formats. For example, typeset documents require different coding methods than laser-printed documents.

Basic Printers

At our model company, four types of printers are available to writers for printing copies of their work:

- high speed line printer
- dot matrix graphics printer
- letter quality printer
- laser printer

The *line printer* is typically used to quickly print a draft copy of a document for editing purposes. Like the graphics printer, it uses dot matrix print, which forms characters out of a series of dots. The cost per copy is extremely low, but so is the print quality.

The *graphics printer* serves to produce charts, line drawings, and accented characters, or draft copies of short documents. The cost per copy and quality is also low, and it is much slower than the line printer. However, it is relatively portable, so a writer can put it on his or her desk if necessary. This is frequently the only way to produce charts and graphics.

The *letter-quality printer* produces characters similar to those found on a good typewriter. It does so by striking characters from a daisy wheel or thimble onto a ribbon and then onto paper. At one time most technical documentation on computers was produced this way.

The *laser printer* produces letter quailty and near-typeset quality documents. Laser printers are often used to replace letter-quality printers. They are cleaner, quieter, and more versatile, offering graphics and a variety of type fonts. Since their cost per page is higher, their use is often confined to final copies of documents bound for a wider distribution.

Most word processors can produce adequate documents with a laser printer, but they often have difficulty taking full advantage of its capabilities. This is true of the word processor in use at our model company.

Part of the problem is the proportional spacing used by many laser printer typefaces. Proportional spacing means that each character can take up a different amount of space, a "w" being wider than an "i," for example. This makes charts and tables difficult to align and anything besides a standard text block extremely laborious to format. Another problem is that selecting the printer's various typefaces can involve extensive tinkering with the printer's internal control program. Sophisticated word processors and desktop publishing systems offer proportional spacing control, called kerning, but at considerable cost and equipment requirements.

Typesetting

Quality standards in the computer marketplace often require typeset manuals. This is true in the case of software packages to be sold in the mass market. The cost is much higher than laser printing, but the competition among software companies requires it.

Most small firms do not have their own in-house typesetting facilities and instead use specialized shops. Typesetting is expensive both for printing and for the time it takes writers to check galley proofs for errors and typographic problems. Fortunately, the errors, uncertainty, and cost of typesetting can be reduced by sending word processing files directly to the typesetting machine via a modem or on a diskette. This prevents the errors a typesetting operator makes while retyping the document. The chapter by Orr discusses transmitting documents electronically in greater detail.

BEYOND WORD PROCESSORS

Spelling Checkers

Word processors often include electronic aids such as on-line spell checking. Spell checking works by comparing each word against an electronic dictionary. Words not found in the dictionary are marked for the writer to check and correct. Unfortunately, even huge dictionaries of 100,000 words found on some systems aren't big enough to contain all of the abbreviations, acronyms and proper nouns found in a typical document. Spelling checkers also can't check the grammar or syntax of written work. For example, repeated words such as "the the," a common mistake, may not be found by an ordinary spelling checker.

For these reasons, on-line spell checking has limited use by technical writers at the sample company, and is confined to the most mechanical type of copy editing. Good editing requires a detailed look at the document as a whole, including checking facts and style. Writers and editors still must read through documents themselves.

Sorting and Mail Merge

Word processors can also offer sorting and mail merge functions. Mail merge typically is used to create form letters used in marketing-related activities. Sorting, originally designed to order address lists for a mail merge, can be used separately by technical writers. It allows writers to create alphabetized lists of error messages, index items and the like.

Outline Processors

As an alternative to the word processor, a number of software companies have developed so-called idea processors or outline processors. These software packages, usually for microcomputers, are developed specifically with the writing process in mind.

Instead of drafting text on a blank "electronic page," writers can list a number of major points with the outliner. Then they go back and either add minor points, producing a more detailed outline, or draft complete text sections. Once the detailed outline, or even the full draft, is completed, they can "collapse" the document so that they see the original points once again. This way they can always keep in touch with the main focus of the document.

Windowing

Another development extending word processing has been the development of windowing systems. Window features built into word processors allow writers to look at two or more sections of the same or different documents on the computer screen at one time. This enables them to compare text sections, much as they would compare two pages of printed text.

Windowing systems working outside of word processors, at the operating system level, offer other advantages. These systems allow one simultaneously to run several computer programs, such as a word processor and a database manager. Each "window" shows a different program operating. These windowing systems allow a writer to run a software package to see how it works, then quickly jump to the word processor to record his or her findings.

Windowing systems have also been developed to put auxiliary tools such as calculators, calendars, clocks and notepads onto the screen, co-existing with word processing systems. Although window systems work best with expanded computers having a hard disk drive and extensive memory capacity, the productivity gains are well worth while.

From an individual point of view, it's good to experiment with different word processing systems. Systems that are easy and comfortable to use can translate into major productivity gains. By contrast, because of their long-range requirements, publications departments need to choose a word processor and stick with it for an extended period. However, this shouldn't preclude the possibility of

investigating alternative systems. The potential productivity gains can make this worthwhile, even though the cost of changing over from one word processor to another can be quite high.

BIBLIOGRAPHY

Bataille, Robert R. Writing in the Word of Work: What Our Graduates Report, *College Composition and Communication*, *33*:3, October 1982.

Bean, John C. Computerized Word-Processing as an Aid to Revision, *College Composition and Communication*, *34*:2, May 1983.

Collier, Robert C. The Word Processor and Revision Strategies, *College Composition and Communication*, *34*:2, May 1983.

Daiutte, Colette A. The Computer as Stylus and Audience, *College Composition and Communication*, *34*:2, May 1983.

Pirsig, Robert. *Zen and the Art of Motorcycle Maintenance*, Morrow, New York, 1984.

PART II

Organizing to Write

Word processing enables writers to rearrange text easily and make corrections of typographic and other minor errors. A clean draft copy, devoid of editing marks, can make more serious problems easier to pick out. As Beck and Stibravy pointed out in the preceding section of this book, those features alone have made word processing appealing to writers. But there is more to writing than clean copy. Using word processing and ancillary programs to accomplish these tasks is the subject of the three chapters in this section.

Studies of the diffusion of innovations such as word processing have shown that innovations are most successful when their use is modified from that expected by their inventors. In his chapter, Strategies for Word Processing in Technical Communication, C. R. Fenno suggests adapting word processing to the entire writing process. For example, he shows how word processing can be used to generate ideas and reorganize them into a coherent outline. Writers then can generate major sections as separate modules and store these items in separate disk files. The word processor can assemble the modules when a printed final copy is needed.

In the second article in this section, Using Electronic Writing Aids as Editors, Robert Krull argues that even a clean draft does not provide an unobstructed view of a manuscript. Technical writers may have difficulty editing because they are inured to quirks in their language and gaps in their logic. They may overuse expressions or words. They may sense the awkwardness or unclear logic of a passage but not the cause of these problems. Krull proposes revised electronic editing procedures to attack content problems hidden behind the manuscript's unmarked face. He argues for editing with two ancillary programs: spelling checkers and grammar checkers.

Krull proposes using a spelling checker to detect technical terms and jargon. This technique is of particular help to technical writers working with scientific or industry specific terms.

Grammar checkers are more elaborate than spelling checkers, analysing language at the phrase rather than the word level. For example, grammar checkers can point out long-winded and hackneyed expressions. Krull argues that creative use of grammar checkers helps writers find complex and subtle language problems.

Both Fenno and Krull write about using word processors to refine the content of manuscripts. In his chapter, Beyond Word Processing: Computers in the Composition Process, F. M. O'Hara describes organizing text and sharing text among writing teams. O'Hara begins by discriminating between the dedicated word processor and the microcomputer controlled by a word-processing program. He argues that using a microcomputer rather than a dedicated word processor affords him additional capabilities. He allows, however, that these capabilities require writers to learn a few additional computer commands.

O'Hara points out that writers using word processing quickly accumulate many manuscript files. He describes efficient ways of naming, duplicating and searching for files. A more elaborate form of text management is provided by database management programs. For example, writers can annotate papers and books to which they refer frequently. A database manager enables them to search through their annotations for information on specific topics, saving them the effort of recalling authors' names and article titles.

Several chapters in the preceding section, notably Wheeler's, have mentioned collaboration among writers. O'Hara describes how writers using different kinds of computers and different word-processing programs can share text. He also describes using electronic mail to manage the relationship among writers.

CHAPTER 4

Strategies for Word Processing in Technical Communication

CHARLES R. FENNO

Recently I asked a group of technical writers and editors about their on-the-job experiences with electronic text processing. The questions covered the following topics:

1. what kinds of word processing equipment they use;
2. what procedures they use when writing and editing on-line;
3. what problems they have encountered; and
4. what solutions they have found for problems.[1]

My motive in asking these questions was both pragmatic and altruistic: as a novice at word processing, I wanted to learn how to transfer my own writing and editing procedures to the computer, and I also wanted to share these professionals' knowledge with other technical communicators who are ready to start working with the computer.

The rest of this Chapter will describe what the professionals taught me, as well as what I learned when I tested their ideas at the computer. By and large, theirs are the good ideas contained in the following paragraphs; mine are more often the lessons learned from mistakes along the way. Both can be instructive.

[1] I'm indebted to the technical writers and editors in the Wisconsin Chapter of the Society for Technical Communication (STC) and in the Air Force Institute of Technology for generously sharing their knowledge.

Learning the Jargon

The jargon of electronic text processing, like the jargon of mystery cults and secret societies, can be a deterrent to the uninitiated. In the case of word processing, however, the code is easy to crack. We need only the eleven terms listed below in order to understand the strategies described by the experienced professionals.

- *copy* – a command that instructs the computer to make a second copy of an entire file or a portion of a file. Useful for duplicating files (an important safety procedure) and for reproducing "boilerplate" sections of documents without retyping.
- *delete* – a command used to erase anything from single characters to entire documents from the computer's memory.
- *file* – electronic storage compartments analogous to file folders in a drawer. May hold chapters of a text, research notes, graphics, or similar items.
- *insert* – a command that appends text to the end of a file or that pushes existing text apart to accommodate additions to a file.
- *mark* – a procedure of enclosing parts of a file in special symbols so that the computer can copy or move the marked parts.
- *move* – used with the "mark" command, a simple procedure for rearranging material in a file by instructing the computer to move marked passages to designated locations.
- *save* – a command that transfers the working copy of a file from the computer screen to permanent storage, usually on magnetic disk or tape.
- *scroll* – a procedure for moving images up or down on the computer screen. Most word processors permit scrolling by single lines or entire screens, and several can be instructed to scroll automatically at a comfortable reading pace.
- *search* – a command that causes the computer to locate a specified string of characters in a file. Some word processors permit automatic replacement of the original string with a second string.
- *software* – computer programs that perform operations on files. Word processors, outline processors, graphics programs, and database managers are popular types of software in technical communication.
- *split screen* (or windows) – the capability of some software to divide the screen into one or more frames, each displaying a different file or different parts of the same file.

CURRENT APPLICATIONS OF ELECTRONIC TEXT PROCESSING

The writers and editors identified six main activities in which they use word processors to produce the *text* (distinct from illustrations) of technical documents. These activities are listed in roughly chronological order.

1. brainstorming
2. collecting information
3. outlining
4. drafting or composing
5. editing
6. producing final copies.

I will discuss them in the same order.

WORD PROCESSOR AND BRAINSTORMING

All methods of brainstorming (and many exist) have one goal: to stimulate the rapid generation and collection of ideas, which are then grouped according to subject. The premise of brainstorming is that creativity flows most freely when the writer is undistracted by matters of style or order.

Composition teachers have recommended brainstorming for years by urging students to jot down—as quickly as possible and without regard to magnitude, sequence, or wording—every idea that comes to mind about a topic. At first glance, the technique may seem more appropriate to a freshman writing classroom than to a technical writing department, but professional writers and editors report many uses of brainstorming. One writer uses it to identify the best information to persuade a potential client to accept a proposal; another uses brainstorming to identify supporting details for descriptions of complex mechanisms.

Word processors are effective tools for brainstorming. Here is the general procedure, distilled from several individual instructions.

1. Use the word processing software to create a new file (often with an optimistic name like IDEAS).
2. Into this file, quickly type every idea that comes to mind about the subject. (Separate the ideas with double strokes of the return/enter key; the word processor automatically handles all other matters of format.)
3. When all ideas are in the file, group or classify related ideas, using the "scroll," "search," and "move" commands described earlier.
   - To begin the classification process, scroll through the file from top to bottom, reading the ideas and mentally identifying *topics* into which they can be classified.
   - Arbitrarily assign a unique keyboard symbol to each *topic*. For example, in a list of ideas about writing, a number sign (#) might represent ideas on editing, and an ampersand (&) might identify ideas on composing. Any symbol that does not appear elsewhere in the text can serve as a topic symbol.
   - After selecting a symbol for each topic, start again at the top of the file, read each separate idea, and classify each by placing the appropriate topic symbol at the beginning of its first line.

4. After all ideas have been tagged with appropriate symbols, use the "search" and "move" commands to group all related ideas.
   - Starting again at the top of the file, instruct the computer to locate and stop at every idea tagged with the number sign, for example.
   - Use the "move" command to extract all "number sign" ideas from the list and group them together, either at the bottom of the list or in a separate file.
   - Repeat the "search and move" procedure for the other topic symbols until all ideas have been grouped by topic. (At this point, of course, the ideas *within* each topic group are still in random order: they will be arranged during the outlining step.)

Testing the Procedure

I found it easy to apply the instructions for brainstorming at the computer—in fact, the procedure is much simpler than the instructions had led me to expect. Each of the commands—"mark," "move," "search"—requires only one or two keystrokes, and each operates quickly and reliably.

However, I encountered one problem from an unexpected quarter when, after concentrating hard and typing rapidly for thirty minutes, I made the mistake of looking at the screen. Seeing several typos, I immediately succumbed to the "Instant Fix" syndrome.[2] This malady arises from the ease of making changes on the screen of a text processor, and its symptoms are an overwhelming temptation to stop brainstorming and start editing. The flow of ideas dries up, and it takes considerable effort to get them started again. Resisting the temptation to edit during a brainstorming session is hard, but the results of resisting are worth the self-discipline required in terms of productivity.

Except for the "Instant Fix" problem, I found that brainstorming at the computer is highly efficient for two reasons: 1) the keyboard captures ideas faster than I can write them in longhand, and 2) the process of electronically arranging ideas into topic groups is demonstrably faster than cutting, pasting, and then retyping paper copy.[3]

DATA COLLECTION

Although technical writers are not primarily researchers, a number of professionals report that their jobs contain a substantial amount of research directly related to technical writing tasks. For example, drafting an effective proposal

[2] David E. Nadziejka of The Institute of Paper Chemistry suggested this apt label for a malady most of us have experienced.

[3] Very recently, new software programs, called "outline processors" or "idea processors" have appeared on the market. Several technical writers told me that these programs are even more flexible than word processors for brainstorming.

can require the writer to collect information on the target audience (often a large organization), on competing products, on previous products and services the writer's company has supplied to the target audience, on the goods or services the writer's company proposes to supply, and on numerous other details. Similarly, technical writers working on complex instructions or product descriptions collect large quantities of technical information on each step of a process or parts of a mechanism. Some of these data come from primary sources, such as technical conferences between writers and engineers; other data come from secondary sources such as trade literature, company files, and other printed sources. Clearly, technical communicators need an efficient data collection system.

Efficient Collection Systems

To be efficient, a collection system must meet three requirements: *accuracy* in recording information, *flexibility* in sorting the notes by topics as the draft is written, and *convenience* in transferring data from the collection medium to the text. Manual notecard systems meet all three of these requirements.

Electronic notecards – When collecting information on note cards, writers often compile source or bibliography cards first, using one 3- X 5-inch index card for each source. Actual data notes are later recorded on 4- X 6- or 5- X 8-inch cards. Experienced writers place a *source label* in one upper corner of each notecard, and a *topic label* in the other. This strategy makes it easy to sort the cards by topics as sections of the report are written, and equally easy to match each notecard to the correct source card for accurate documentation or for followup if additional information is needed. The only constraint is that a new notecard must be used for each different source and for each different topic within each source, so that the system has maximum sorting flexibility. During drafting, some writers actually tape or paste the notecards onto the draft pages for typing into the final copy.

Since notecards are accurate, flexible, and (reasonably) convenient, technical writers have used them as models for electronic data collection systems.

Writers can create electronic notecards with either of two kinds of software. The most effective packages, called *database programs*, store information in a format that can later be sorted and retrieved in any number of ways. Standard word processors can also be used for data collection but, as I discovered, they are considerably less efficient for this job than database programs.

Databases – To start taking notes with database software, the writer first creates two screens or templates. The first resembles a customary source or bibliography card, with blanks to hold all of the information necessary to identify sources accurately. The other template resembles a paper notecard, with lines for source identifiers in one corner, for one or more topic identifiers in the other, and with several blank lines to hold the notes themselves. The writer creates these empty templates only once: the computer provides copies of

them each time a new "card" is needed. Notetaking can begin as soon as the two templates are ready.

At this point, the physical distance between the computer and the sources becomes important. If the database software is on a personal computer, the writer can carry the computer to the sources (for example, a technical conference or a library) and take notes there. Or, if the database is on a stationary computer, a portable terminal located near the sources can transmit notes to the computer through a telephone line. If all else fails, the researcher must move the sources to the computer—surely the modern equivalent of bringing the mountain to Mohammed.

Information is entered on the electronic notecoards exactly as on paper cards. When all information has been collected, the database program performs one of the jobs it does best: sorting. After weeks of collecting information for a document that contains several dozen topics, the writer issues a few instructions and the database sorts its contents by source or by topic and prints the results. In addition, many database programs contain their own word processors (or are compatible with external word processors), which can move the contents of the electronic notecards directly into the draft of a report without extensive retyping. In the same way, entries on the source cards can be sorted and listed by any of that template's entry blanks (e.g., source or date), and the entire bibliography can be alphabetized and printed by the database or a compatible word processor.

Word processors – The reason that word processors are less suitable than databases for notetaking is that word processors cannot match database flexibility in sorting. To take notes with a word processor, a writer must choose between creating a new file either for each *source* or for each *topic* of information. Creating files by topics produces useful topical printouts of notes, but entering information by topics requires switching from one file to another every time a source contains information on more than one topic—a time-consuming procedure.

Standard word processors are slightly more convenient for source or bibliography cards than for notecards—but only slightly. A few word processors (and some dictionary programs) allow the user to alphabetize lists that contain only one word per line. Thus, while they cannot alphabetize a complete source list, these programs *can* alphabetize a list of single sources (for example, authors' last names), thus providing a useful quality check on manual alphabetization of long bibliographies. However, since databases can alphabetize any of their contents, including complete bibliographies, word processors are still comparatively less efficient.

I finished my notetaking test at the computer with a new appreciation of the importance of having the right software for the task at hand. Database programs make the computer a powerful note collection tool, while word processing

programs—at least the ones that the professionals recommend and that I tried—offer only limited support for notetaking. Both kinds of software deserve a place in a technical writer's office.

ORGANIZING OUTLINES

Organizing the outline of a technical document requires the writer to make rhetorical judgments related to the document's purpose, audience, and content and, simultaneously, to consider non-rhetorical constraints, such as company policy or the requirement to follow prescribed formats. Since a document's unity and coherence are first established during this organizational step, some writers consider this the most important—and difficult—step in the communication process.

Obviously, the computer cannot make judgments, but word processing software arranges material easily and quickly to reflect the writer's decisions. As a case in point, I printed a paper copy of the IDEAS file created during brainstorming, and then numbered the main topics on the paper copy to indicate how I wanted to arrange them, in this instance, chronologically. It was then an easy job to use the word processor's "move" command to put the topics in their proper order in the IDEAS file. Experienced writers often omit the paper step and build the outline directly in a copy of the IDEAS file, a more efficient procedure than the one I used. In either case, the product is the beginning of an outline for drafting.

The next step is to refine the outline by using the word processor to arrange and polish the ideas *within* each main outline topic, inserting, deleting, and moving material as necessary. This electronic "cut-and-paste" is considerably faster than manual methods, since it requires no retyping. When the outline is in final form, drafting can begin.

COMPOSING ON WORD PROCESSORS

One technical writing colleague compares the process of composing on the word processor to working on a "very forgiving typewriter." As he says, the word processor accepts text as fast as we can type, allows on-screen corrections, and produces a final copy that hides the false starts and dead ends that we encountered along the way. This uncharacteristically rhapsodic statement was echoed by other technical writers and editors I talked with: measured in terms of efficiency, electronic composing and editing are the best things to happen to our profession since the invention of ballpoint pens and transparent tape.

In fact, it is the efficiency of word processing that prompted one technical writer to suggest that novices at word processing should remind themselves of the distinction between composing and editing. The goal of composing is to "flesh out" the ideas arranged in the outline. Viewed this way, composing is a

creative process in which the writer cultivates a kind of "rhetorical inspiration" to find the best ways to communicate technical material to its intended audience. This creative development can be impeded if the writer stops to labor over every turn of phrase in the first draft, exactly as the "Instant Fix" syndrome impedes brainstorming. One way to keep the ideas coming is to compose "warts and all" in the first draft, giving highest priority to the critical and creative selection of supporting material for each main idea in the outline, and to the task of achieving tight unity and coherence. Final polishing of the text will occur in the subsequent editing step.

General Composing Steps

The steps for composing with a word processor are essentially the same that we use when composing in other media. Here is a generalized outline of the process many writers follow, regardless of the medium:

1. Divide the outline into sections at its major topics. (This division permits incremental or modular writing, and is especially important when several writers are working on the same project.)
2. Produce text for each section of the outline, working "top down" from the largest ideas to the smaller details until each idea is adequately developed.
   - Indicate the places of necessary documentation or text notes by writing them directly into the text for later placement by the typist or typesetter.
   - Indicate the places for tables and graphics by skipping several lines of text and jotting down the content and approximate size of each.
3. When drafting is completed, arrange the modules of text into their correct sequence to obtain the complete draft.

Special Adaptations

When these three steps are transferred to a word processor, a few changes are necessary because of the special capabilities of word processors and the need to ensure the document's safety. The following are some of the most important adaptations.

Dividing the outline – When the outline is divided in the word processor, each section is stored in a small electronic file with a descriptive file name. The utility of small files is evident when we consider how the word processor operates. The computers which underlie word processors manipulate text files by internal programs that are invisible to the user. These programs can manage smaller files faster and more accurately than larger files. In addition, small files minimize the amount of text that would be lost in the rare chance that the computer malfunctions while a long file is open.

Descriptive file names are essential as small files proliferate. A writer assigns a name to each file when it is created. At the beginning of subsequent work sessions, the word processor lists all files on the screen so that the user can select

appropriate ones for further work. Non-descriptive file names such as TEXT1, TEXT2, and so on, give very little help when the writer or editor needs to locate a specific file containing, say, test results or cost data. Descriptive file names such as COST, TESTS, or IDEAS permit easy identification and management of many files.

Footnotes and endnotes – Footnotes and other detached forms of documentation are time consuming when a draft is composed on paper. Regardless of where the notes are placed in the draft originally, subsequent editing usually changes their order. Word processors provide an efficient solution to this problem. The writer types the notes directly into the body of the document at whatever place they should appear in the final document and marks them with a special symbol. When the draft is later formatted for printing, the word processor's "search" command finds the notes, and the "move" command places the notes into their final positions as footnotes or end notes. Indeed, one kind of word processor automates all of these processes during formatting.

Tables – Tables pose another kind of problem during drafting or editing, since they are time consuming to retype each time they must be moved and yet there are sound rhetorical advantages to being able to see them in place during composition. With a word processor, each table or computer-generated graphic is created and stored in its own file. As the text is drafted, simple commands instruct the computer to copy a needed file into the text. Since the original table remains in storage, it can be recopied at other locations as needed when the text is edited.

Safety – Safety of the files becomes increasingly important as more and more text is drafted. In addition to the safety achieved by using several small files rather than one large file during drafting, writers usually take the precaution of keeping duplicate copies of valuable files. If possible, these duplicates are stored on separate tapes or disks from the originals so that even if one entire set of files is lost (for example, an entire magnetic disk), the work can be restored from the copy. By and large, computers are highly reliable, performing day after day without a problem, but occasionally a problem can occur—an electrical blackout or a mechanical failure in the system—that destroys any file in use at the time. To minimize even this remote chance of loss, simple safety precautions are well worth the few minutes they take at the end of a day's work.

EDITING THE DRAFT

One of the first lessons every technical editor learns is how many different tasks are summed up in the deceptively simple term "editing."[4] In its largest sense, editing encompasses all of the activities required to make a document

[4] For example, Mary Fran Buehler describes nine types of editing activities used at the Jet Propulsion Laboratory, in Defining Technical Terms in Technical Editing: The Levels of Edit as a Model, *Technical Communication*, *28*, pp. 10-15, 1981.

rhetorically effective—that is, able to do its intended job for its intended audience.

Some of these editing activities are handled the same way on paper copy or on the screen of a word processor. For example, ensuring that the content of a handbook or manual conforms to local policy is an editing requirement that is independent of the medium used to produce the document. On the other hand, several editing activities *are* affected by the production medium, especially activities dealing with a document's content, language, and organization. The professionals I talked to described their strategies for on-line editing, beginning with their preparations for editing.

Editing Preparation

Two important preparations for editing on a word processor are *rechecking the outline* and *assembling support files.*

Rechecking the outline – Rechecking the outline is essential when the editor is not the person who composed the text, and it also provides a useful quality check for authors editing their own drafts. The recheck ensures that both the main topics and the ideas within the topics are in their best order, and that adequate and appropriate supporting material is present for each point in the outline.

During the recheck, the editor scans the outline file to ensure unity, coherence, completeness, and emphasis in the structure. If changes are needed, they are easily made with the word processor's "insert," "delete," and "move" commands, in the same way that the ideas in the brainstorming file were rearranged at the end of the brainstorming process. (Changes to the outline will be duplicated in the draft text files as a part of the editing process.) Finally, a copy of the revised outline is printed to serve as a guide for later editing steps.

Assembling support files – One of the distinct advantages of composing and editing at the computer is the fact that other kinds of software can be used in conjunction with the word processor to support the editor's work. Dictionary, thesaurus, and usage programs are available to perform electronically the same functions as their paper counterparts. If an editor expects no more from support programs than they are intended to deliver, these commercially available programs can save time during routine copyediting.

A few innovative editors told me that they also used the word processor to create their own support files for editing and for training new writers in their departments. Examples include files that contain lists of both proscribed and required words in company publications. By storing these lists in on-line files, a writer or editor can update them easily.

When these and other kinds of special files have been assembled, editing begins.

Managing Electronic Editing

For brainstorming and notetaking, it is easy to extract general steps that most technical communicators follow. In contrast, editing styles are too individualistic to distill into a single list that would be comfortable to every editor. Thus, here I'll describe only those differences between editing paper copy and editing with a word processor that are independent of personal editing style. The following are some of the professionals' strategies for managing on-line editing efficiently.

Access problems – When we edit paper copy, the text is physically accessible on the desk. An editor can turn the pages backward and forward to check structural unity, coherence, completeness, and emphasis in passages that are several pages, or even chapters, apart. In contrast, many computer screens display only *twenty-four lines of text* at one time. Overcoming this limited or "micro" view of the document is, according to professional editors, *the* most important adjustment required by on-line editing.

One apparent solution to the limitations of the small screen would be to print the document and to edit the paper copy. However, even if time is available for this extra step, it is not necessary: modern text production provides many examples of "paperless" editing. Today, the text of large newspapers is composed on-line by the copywriter, edited on-line by the copy editor, and fed to typesetters by the computer. In another application, an author of a technical writing textbook in Texas told me recently that he and his New York editor transmit drafts and revisions to each other on magnetic disks instead of paper to save time (the publisher even supplied the computer). And not long ago, a technical writer in Wisconsin described to me his company's recent implementation of Computer Aided Publishing (CAP), where documents move from outline to finished printing without paper intervention. In situations like these, editors use their word processors' capabilities to improve on-line document access, including the following.

Special text markers – One popular word processor allows the user to insert as many as nine special markers in the text, so that only two keystrokes are needed to switch the screen instantly from one marked passage of text to another. If these marks are placed at the beginning of passages that need to be compared during editing, the effect is similar to flipping paper copy backward or forward.

Split screens – The split screen feature described earlier in this chapter also compensates for the limited access problem. A split screen displays two sections of a file simultaneously, permitting easy comparisons. The editor can scroll text in either window of the split screen, as well as transfer text from one window to the other.

Scrolling – The scroll command also allows an editor to move forward or backward through the text files, reading successive topic sentences or paragraphs,

a useful procedure for checking unity and coherence, and for verifying that headings and subheadings are consistent.

Automatic text exchanges – Automatic text exchanges compensate for the small screen's limited access by permitting changes to text that is not currently displayed on the screen. These changes are managed by the word processor's "search and replace" command. To illustrate how this feature works, one editor described a project in which technical writers in a large company had to produce descriptions of a new product before the legal department had secured its trade name. The writers drafted the documents by referring to the product with the code name "newthing," an easy-to-type nonsense phrase that would be unique in the document. When the trade name was announced, the project editor used "search and replace" to exchange the trade name for the code name throughout each document—a chore that required about ninety seconds.

Advantages of Electronic Editing

In balance, the writers and editors I talked with believe that the advantages of word processing considerably outweigh the inconvenience of having to adapt individual editing techniques to the electronic medium—especially when we consider that the word processor links composing, editing, and printing into a more efficient continuum than we can obtain from any other method of text production.

However, one small problem should be mentioned. Because the process of editing includes many different activities, it places many demands on a word processor. Unfortunately, no single word processor is likely to contain every feature that the editor might want. For example, the split screen processor that I tested does not permit the use of special marking symbols. Similarly, the popular word processor that provides special marking symbols does not allow the editor to restore deleted text, a useful feature of the split screen processor.

There are two important lessons here, I think. First, the technical communicator who is in the market for word-processing software should carefully examine the capabilities of each package and, if possible, talk with experienced users about the suitability of specific software for technical writing and editing.

And second, when the word-processing software has been installed on the computer, the writer or editor should master its features thoroughly, since thorough knowledge fosters ingenuity, and human ingenuity can make a word processor rise above its own limitations. The creative applications described in the preceding paragraphs were developed by communicators who invested enough time to learn the tools of their trade.

PRODUCING THE FINAL DOCUMENT

After the rigors of editing, the sight of page after page of finished text emerging from a printer is particularly satisfying. At the end of the editing step, only one task remains before printing can start: the text files must be formatted.

On-Screen Formatting

With some word processors, the editor designs the finished page on the screen. Margins, indentations, and line spacing are established exactly as the editor wants them to appear on paper. Except for special non-printable characters imbedded in the text to instruct the printer to underline, boldface, or even to change print color, type size, or type font, the image on the screen will be reproduced on the paper copy.

Background Formatting

Other word processors handle formatting as a two-step process. In the first step, special command words are typed directly into the text file. Each command word performs a special job: one might create an indented list; another might block and center text for long quotations. On the screen, the text itself appears just as it did during composition, with only an occasional adjustment of line spacing required to indicate which passages are to be affected by the command.

When all of the command words are in place, the editor closes the files and, as step two, instructs the word processor to begin background formatting. The word processor produces a second copy of the file, formats it according to the command words, and sends it either to the screen for the editor's review, or to file storage for later printing. The original text file is preserved unchanged.

Formatting Tradeoffs

Each method of formatting has advantages and disadvantages. On-screen formatting requires more preliminary typing. For example, tabs, margins, and other details of indentation must all be set by keystrokes very much as they would be set or changed by levers and keystrokes on a typewriter. However, this method of formatting lets the editor see exactly where page breaks will occur, a major advantage.

In contrast, the off-screen method automatically handles many details of formatting, even supplying numbers for indented lists, but off-screen processors do not show where pages will end in the printed copy, an inconvenience that requires the editor to re-open files repeatedly to adjust page breaks.

In formatting, then, as in other word processing features, the user must make tradeoffs. Whether a particular method of formatting will feel comfortable to the user is determined in part by the characteristics of the software and in part by the user's preferences. "Try before you buy" is the professionals' recommendation.

Printing the Document

Since word processors count pages and automatically insert page numbers during printing, the formatted text files must be linked together to print as a continuous document. Processors that format on-screen usually accomplish file

merger by the "copy" command: the text files are copied one after another into a single large file. Background processors link the files by means of a small program which the user creates. When the files are assembled in proper order by either process, printing can begin.

Certainly one of the factors that influence the communication effectiveness of technical documents is the texts' appearance on the printed page. In turn, appearance is affected by such varied factors as format, graphics, and printing quality—all of which a publication manager must evaluate and control. However, our topic in this essay is the use of word processors to *prepare* technical documents for publication, and we are at the end of that process.

SUMMARY

I concluded my personal test of word processing convinced that this electronic technology is a valuable tool for technical communicators. Like every good tool, word processing is sufficiently flexible that it can be adapted to its user's customary work strategies. And most importantly, it can make those strategies more *efficient*.

Can it also make us more *effective* writers and editors? Effectiveness, I suspect, is the product of three characteristics I observed in all of the technical communicators who contributed to this essay:

1. a desire to learn more about the principles that underlie communication,
2. an eagerness to sharpen personal skills in applying those principles, and
3. the dedication to do the best job possible with *reasonable* expenditures of company and personal resources.

Continuing education and on-the-job experience will help us in the first two areas. Electronic word processing can help us with the third.

CHAPTER 5

Using Electronic Writing Aids as Editors

ROBERT KRULL

The report was over a hundred pages long and had to be at the printer in two days. It had to be to the federal government within a week. A final edit and I was finished; but that last edit could be difficult. I had read this material so many times that I no longer could tell which sections covered which topics, whether any of it made sense, and whether words were spelled correctly.

You have no doubt been in situations like this yourself. What I could have used was an editor who was familiar with what I was writing about, could edit for content as well as style, and would laboriously check for typographic errors. Most writers, myself included, don't have such an editor available. However, we may have access to electronic writing tools that perform some of these functions. My word processor would let me make the changes easily once they were identified.

WISHFUL WRITING AIDS

The Report Writer

A Utopian electronic writing aid would do the whole job by itself. It would decide what to write about, structure the content, and fill the structure with appropriate text. Fortunately this writing aid does not exist. I say "fortunately" because there would be fewer jobs for writers if this tool were available. However, writers still can use electronic support.

Electronic prewriting tools have been used most in teaching writing to high school and college students. Business versions of outlining aids have been promoted as enabling writers to concentrate on the structure of a piece before refining the writing. Both tools help with the first two stages of the Utopian

report writer—deciding what to write about and building structure. Prewriting and outlining aids have been discussed in detail in books on teaching writing with computers (Halpern and Ligget, 1984; Shostak, 1984; Wresch, 1984). I will not discuss them further.

Writing the first draft and determining which parts need changing requires other tools. The advantages of word processors for drafting and revising has been covered in other chapters of this book. Deciding what needs revision is another matter. This chapter examines two additional augmentations to word processing—spelling and grammar checkers. It delimits what these tools can and cannot do. It also suggests a writing strategy that minimizes writer effort and maximizes the help these checkers can provide. Finally, it describes a way to use a word processor's search and block move functions to enhance a text's logical flow.

The Perfect Spelling Checker

All writers, and technical writers in particular, want to eliminate typographic errors, misspellings, and erroneous placement of words. For creative writers such errors reduce the emotional effectiveness of their texts; for technical writers they produce calls from irate customers, unnecessary revisions of documents, and potential product liability cases.

The ideal electronic spelling checker should detect incorrectly spelled words, words that don't conform to an accepted technical vocabulary (including acronyms and jargon), and words that, while spelled correctly, are incorrect in context. Most electronic spelling checkers compare each word in a text to a list of standard English words. If the program encounters a word not in its list, it will notify the writer. This does not mean the word is incorrectly spelled. The word can be a properly spelled standard English word, a technical term, or an acronym but may not appear on the short (20,000 to 50,000 word) list which the spelling checker consults. Most spelling checkers can be expanded, as I explain in a following section.

Writers would like to detect out-of-context words. However, since spelling checkers really only compare words in a text to a list of words, they generally do little, if any, assessment of the relationship of individual words to other words. A few spelling checkers detect double words. For example, some spelling checkers would flag the following sentence:

I don't like your your writing.

It would flag the second "your" in the sentence because it can easily be programmed to find double words. Most spelling checkers would not flag the following sentence:

What do your think of his writing?

The spelling checker would have to be sophisticated to guess that the word in the sentence should be "you" not "your." Few spelling or grammar checkers can

correctly interpret this sentence. That means some person must read through texts at least once to assure their being free of erroneous words. The perfect spelling checker does not exist; but in a later section I describe how list-based checkers may be used to advantage.

The Perfect Grammar Checker

The ideal human editor detects poor word choice, cliches and hackneyed expressions, and awkward sentence construction. Electronic grammar checkers available today can handle the first three problems and can help with the fourth.

Electronic grammar checkers, like spelling checkers, compare sections of text to lists. The grammar checker's lists contain cliches, hackneyed expressions, and frequently misused words and phrases. Their lists also may contain frequently used, but awkward phrases such as "a lot of" and "a number of." Most grammar checkers help writers by suggesting other ways to say the same thing. A few sophisticated grammar checkers may also detect compound sentences and conditional clauses. Once the checker finds such sentences, it is up to writers to determine if and how they should be changed. In a following section I discuss a strategy for taking full advantage of the grammar checker's limited abilities.

The Third Eye

A final service from human editors is to assess the logical flow of our content. Unless we are writing to a formula, electronic writing tools are little help here. For example, several journalism schools use computer programs that determine if facts in stories are arranged in the proper order and given appropriate emphasis. Special purpose programs can perform this function because instructors first specify important facts and the order in which they should appear. The program then compares student-generated text against this fact list. Unfortunately, technical writers working on new documents do not have someone to generate fact lists for them and a program needs the list to work from.

However, several of the electronic writing aids already discussed can help technical writers obtain a third-eye view of their text. I explain how this is possible as I describe a strategy for using each writing tool in the following pages.

In summary, we don't have the ideal writing tool at our disposal. We do have outlining aids, simple spelling and grammar checkers, and the opportunity to use these tools creatively.

WHEN WRITING AIDS HELP

Limited though these tools are, they can be helpful under certain circumstances. Recall my writing my federal government report. I met every condition.

1. *When we're under deadline.* Technical writers work under deadline. Sometimes the deadlines are very tight. For my government report I had only two

days to edit about sixty pages and produce a camera ready copy for the printer. Human editors cannot make quick, multiple passes through the text checking for spelling and grammar problems. You may be able to find someone to do it once or twice, but not many times. Electronic writing aids are tireless and very fast.

2. *When we are too tired to read more.* Writing and editing take energy. Sometimes we become sloppy and willing to accept substandard work when deadline pressure combines with fatigue. An electronic tool can do the brute-force analysis of our text, saving our energy for the more creative editing tasks. For example, when my government report reduced me to speaking jibberish at dinner, my spelling and grammar checkers were nit-picking through my text, looking for the errors I could no longer see.

3. *When we have a formal style guide.* A formal style guide makes human text checking more difficult: there may be so many items to check for. But a formal style guide makes machine checking easier. The more concrete the style guide, the easier checking becomes. Acronyms, jargon, stock phrases, and outlawed phrases can all be added to electronic lists. Once specified, they will be found by electronic tools. Unfortunately, I had no style guide for my government report. I have been slowly adding words and phrases to my own electronic tool kits.

4. *When we are unsure why our writing isn't working.* After repeatedly rereading our own writing we may not be able to tell what its problems are. Ideally we should put the text aside and return to it later; but deadlines may not let us do so. With my government report I found my electronic tools gave me unexpected perspectives on my text. They pointed out quirks in word choice and phrase making, overuse of some expressions and technical terms, and, indirectly, problems in logical flow.

5. *When someone else has written what we don't want to read.* We search for ideal human editors and don't find them. Worse, other people engage in the same search and choose us as candidates. Most of us don't want to add other people's deadlines to our own. Where we cannot detect problems in our own writing, we are all too able to find them in other people's. We find so many that our editing task grows into re-writing. Wouldn't it be pleasant to turn this task over to someone, or something else? You can. Refuse to edit by eye that which has not first gone through electronic scrutiny.

USING A SPELLING CHECKER TO ADVANTAGE

Expanding the Basic Spelling Checker

Only a few spelling checkers are part of a word-processing program. This type of checker operates while a writer is using the word processor and calls the writer's attention to suspect words. Most spelling checkers run separately from

the word processor. These checkers analyze text once it has been entered by means of the word processor. Both types of checkers do the same task. As I have mentioned, they compare each word in a text to a list of acceptable words derived from standard English, an industry-specific technical vocabulary, or a company-specific list of acronyms and jargon. Generally, the industry- and company-specific lists are kept in separate user dictionaries. This procedure is advantageous to the technical writer.

If the spelling checker finds a word not included in one of its lists, it calls the writer's attention. The writer then has several options: correct the spelling, leave the word as-is so that it will be flagged again by the spelling checker, or add a correctly spelled word to a user dictionary. Most writers would immediately correct an incorrectly spelled word. Some technical writers would be tempted to add all technical terms and acronyms to their user dictionary. That way the spelling checker will detect when these terms are incorrectly spelled but won't call the writers' attention every time the terms are used. I would argue against this procedure.

My reason for doing so is that readers of technical material, generally speakers of standard English, also may not have technical terms and jargon on their mental lists of acceptable words. When they encounter many terms with which they are unfamiliar, they understand less of the text. If technical writers don't add all technical terms, jargon, and acronyms to their main spelling checker dictionary, they can use the checker as a jargon detector to indicate where readers are likely to misunderstand the text.

Ideally, writers should have a separate dictionary, called a user dictionary, for problem words. Most spelling checkers provide such a facility. They can then run the spelling checker once using standard English dictionaries and once using their expanded technical English dictionaries. The first pass detects jargon; the second pass flags incorrect spellings.

Examining Word Frequency and Abstraction

Spelling checkers can also compute how often words occur in a text. Most writers like to have variety in their writing and can use word frequencies to see if they tend to overuse some words. Technical writers may be more concerned with using too many difficult or abstract words. The frequency list can show how many times a text contains polysyllabic words and nominalizations.

I found this feature useful when editing my government report. A frequency analysis of the seventy-two sentence, 1324 word opening section showed the following:

- I had used 412 different words.
- I had used the following words 10 or more times:
  I (22), children (17), information (15), part (12), production (22), program (16), programs (11), research (23), section (10), television; are (10), be (16),

have (11), is (25); a (31), about (15), and (26), for (13), how (21), in (29), of (48), the (80), this (16), to (46).

What do these frequencies tell us about the text? It would be useful to have a standard against which to judge them. Writing analysis tools such as "The Writer's Workbench," developed at Bell Laboratories, contain statistics for documents judged to represent good writing. They will even summarize differences between the text and model documents. I didn't have a model and had to interpret the statistics myself.

Pronouns and nouns constitute the first group in the list. "I" appears twenty-two times, or about once every three sentences. That may be too egoistic. The nouns children, television, program(s), and production also occur many times. From that you may be able to guess that my report dealt with production of television programs for children. My frequent use of these words may indicate that I was consistent in sticking to the topic of the report. On the other hand, the title page also told readers that this was the report's topic. I might have been able to assume this knowledge and more fruitfully discuss its details.

The remaining nouns—part, section, information, and research—are abstract. The first two refer to the report's organization and may be reasonable given the opening section's purpose of telling readers how to use the report. Information and research, however, while referring to the work on which the report is based, are not specific. They may not have helped reader understanding.

Three of the four verbs in the list are forms of the verb "to be." Many style guides argue this weakens writing.

The last group in the list includes many prepositions and articles. Is this good or bad? Further analysis by a grammar checker, which I will discuss shortly, clarifies that this shows an overuse of prepositional phrases.

In summary, the frequency counts help me detect the overuse of abstract words and the verb "to be," and, perhaps, too many prepositions. This probably argues for an additional edit. Missing from the list reproduced here are the 400 or so words that occurred two or fewer times. Do they add enough variety to the text to save it from a rewrite? The electronic spelling checker doesn't help with such decisions. Either a comparison against model documents or, preferably, a human editor would be required to judge.

For example, the frequency list for another section of the report showed that the word "autocorrelation" appeared often. It is both polysyllabic and an obscure statistical concept in which my readers were probably uninterested. When I examined my report to see where I had used the word, I found that my tone became too technical in every case. Rather than finding a synonym for autocorrelation, I decided to rework those sections entirely. I think my report was more usable as a result.

Writing with a Spelling Checker

Writing styles vary considerably. You may find that the writing procedure I am going to advocate doesn't meet your needs. Modify it to your advantage. Also keep in mind that your writing procedure may change as you gain experience with electronic writing aids. Here is the procedure that most helps me take advantage of my spelling checker.

1. Write the first draft with your word processor as you would ordinarily. If you are using a spelling checker that examines your text as you type it, you may want to turn it off for the initial draft. Otherwise you may find yourself interrupting your writing to correct spelling.
2. Edit major structural problems before running the spell checker. Fixing the spelling of text you are going to throw away is a waste of effort.
3. Run the spelling checker as part of the second edit. First run it using all your technical dictionaries, insuring technical and company-specific terms are spelled correctly. You can add new terms now if they are not part of your special dictionaries. Also, correct obvious misspellings right away.
4. Run spell check again, this time using the standard English dictionaries. Check to see where you use acronyms and technical terms (remember the spelling checker will flag them for you since they are not part of standard English). Are all defined? Do they all appear in a glossary?
5. Run the frequency count option of your spelling checker. Check the listing for excess use of long words, technical terms and acronyms. If you overuse a word, you could try to find an alternative. If you use too much jargon, you may not be understandable to readers. Also compare frequencies to your preceding texts and to model texts. Are you getting better or worse? Is your writing compatible with document libraries of which this particular manuscript is a part?
6. After the final edit, rerun the spelling checker again with all the dictionaries to detect problems introduced during editing.
7. Read through the final draft to make sure it does not include errors that will elude electronic writing aids (e.g., words correctly spelled but incorrectly placed or used).

USING A GRAMMAR CHECKER TO ADVANTAGE

A grammar checker is not integrated into a word processor program; it runs separately on text generated by means of the word processor. Depending on its sophistication, the grammar checker can detect trite and cliched phrases and those disallowed by a style guide, compute readability and other style statistics, and offer language alternatives to writers. Grammar checkers won't analyze logical flow and cannot tell technical writers if their content is appropriate to their readers' tasks. However, in analyzing problems pointed out by my grammar

checker, I found that often the real problem is in adjacent sentences. In other words, the grammar checker finds symptoms of larger writing problems. I will explain how I use this feature to full advantage.

Writing with a Grammar Checker

1. Write the first draft quickly with your word processor, knowing poor usage can be detected later.

2. Before editing the first draft, run the grammar checker to see if its language difficulty falls into the range acceptable to your audience and if it meets style guide criteria. For example, you might look at readability scores, number of subordinate clauses, number of passive sentences, number of prepositional phrases, and so on.

The objective is to determine the direction of your first edit, rather than trying to hit a specific target such as a reading grade level. By direction I mean whether your writing should be made more direct and precise or if you have met that goal. If you have, you can concentrate on tuning specific sections of text instead of reworking its whole tone.

For example, the descriptive statistics for the first section of my government report showed the following:

- 1324 words were divided into seventy-two sentences for an average sentence length of nearly nineteen words. If I had a style guide, I would know if these sentences were about the right length for my audience of television producers. My longest sentence, sixty-one words, probably is too long for any audience.
- Well over two-thirds of my sentences, fifty-nine of them, contained forms of the verb "to be." I didn't need a style guide to know that this needed rewriting. For example, the phrase "children are paying attention" could be shortened to "children attend." The grammar checker advocated that a change be made, but not how it should be made. I had to decide that.
- My seventy-two sentences also contained 172 prepositional phrases. A style guide could have told me what is typical for reports of this type. However, two and a half prepositions per sentence is likely to be too high. I needed to tighten my writing so that sentences conveyed their meaning more directly.

My grammar checker is not sufficiently sophisticated to catch passive verbs and subordinate clauses but, had it been able to, I am sure it would have found a number. This numeric description of my text told me that my writing was obtuse; it was not as direct as my journalism school training had taught me. The advantage given me by the checker is that it told me where I could improve my writing.

3. Next look at the checker's suggestions about specific sentences. Most checkers allow you to go through your text line by line, pointing out problems and, either marking the text for later revision with your word processor, or letting you make revisions immediately.

If the solution is obvious, you can make changes when problems are flagged. For example, the grammar checker pointed out that "affect," "effect," and

"already" are frequently misused and cited their correct usage. Since I had used them correctly in my report, I could simply ignore the checker's advice on this point. Had changes been required, I would have made them during the second content edit. I prefer the latter.

In editing my government report I often had to look one or two sentences ahead or behind the problem sentence. For example, I seem to overuse vague modifiers like "quite," "fairly," and "rather." This is due to my being careful not to overstate the significance of research findings, as in "fairly strong results." My rereading showed the solution was either to rewrite surrounding material or simply to drop the modifier. I decided the most effective solution was to do both. Since I had used words like "fairly" to compensate for excessively strong language, simply removing the words left the wrong impression. Most effective was my precisely stating what I meant.

4. Finally, you should rerun the grammar checker to find new problems you might have introduced and to see if the style statistics indicate your more closely matching the needs of your target audience. I think you will find the strategy I have suggested more successful than trying directly to achieve a particular readability score.

My procedure may seem complex. You do have to read the spelling and grammar checkers from disk and have them work through your text. In my own work I don't bother with electronic aids for documents shorter than 10 pages. For longer documents, running electronic aids is faster than cajoling reluctant editors.

For example, it takes me at most one minute to load either of my electronic aids. Either checker requires about one minute to make a first pass through a twenty-page document. Printing these initial results takes another minute. At this point I have a list of possible misspellings and style statistics such as the number of words, sentences, prepositions, uses of the verb "to be," and so on. So within ten minutes I can obtain an overall assessment of my text. I could obtain a printed list of word frequencies in another five minutes.

How long it takes to make suggested changes depends on the number of problems the electronic checkers find. For example, my grammar checker processes about a problem-free page in a couple of seconds. When it does encounter a problem it stops, suggests a change, and waits for me to make a decision. It may take me a minute or so to make up my mind. If I have one problem per page, it could take twenty or more minutes to go through a twenty-page text. How long would it take me to read through the manuscript in detail? Probably considerably more than twenty minutes. The time advantage for the electronic checkers grows as the text analyzed gets longer.

CATALOGING LOGICAL FLOW

Rereading our writing helps us detect problems in logical flow. Unfortunately, after several edits, we can no longer remember where we discuss specific topics and whether we have deleted some material. Reading through our text again

word-for-word can make us only more confused. Most grammar checkers won't analyze this aspect of our writing. However, I have found two strategies for using a word processor to help us analyze logical flow.

Search Functions and Logic

The search-and-replace feature has already been discussed in other chapters. Typically it is used to change systematic misspellings. However, you can use the search function to check logical flow as well.

1. After the first draft, get a printed copy of your text. Use the word processor search function to find each major topic. Mark these locations on the printed draft. Use colored ink, underline the words, or mark with symbols to discriminate among different topics.

2. After the major topics are marked, scan the text to see if all the supporting material for each topic is located in one area. If any topics are spread throughout the manuscript, your readers may have difficulty following your logic.

Regrouping sections to collect topic-specific material will make your text more readable. First, you need to define the revised structure. Then, you can again search through the text using the word processor's search function. Finally, you can rearrange blocks of text.

Index Generation and Logic

Some word processors will automatically create an index if you tell them which words or phrases to include. The index is useful to readers because it provides direct access to specific information. The index is useful to writers because it helps check logical flow. Let me explain how you can do so using your word processor's indexing capabilities (if your word processor does not construct indexes you cannot use this method).

1. Go through your text marking major and minor topics with your word processor's indexing function. Limit the number of marked locations to the number of sections you would be willing to move.

2. Have the word processor generate the index. Then look through the page references in the index. If you find widely separated page references for one or more topics, your text's logical flow may not be coherent. Consider rearranging your manuscript.

3. After writing your final draft, have the word processor regenerate a full index. Both major and most minor topics should now be covered in smaller page ranges.

As with the grammar checker, index construction as a means of assessing logical flow seems time consuming. I wouldn't bother if I weren't already constructing an index. Using the search function, on the other hand, requires little time under most circumstances. My word processor lets me store about eighteen pages of text in memory at one time. The search function can process the entire

document in about two seconds. If the word processor finds many instances of my search string, this can take much longer. The reason again is that I have to decide what to do rather than that the computer takes a long time doing its task.

The one condition under which word processors search more slowly is if they only hold one page of text in memory and must read in more text from a disk or other storage medium for every new page searched. Such word processors are more appropriate for secretaries who type short texts without analyzing them instead of writers who must generate and edit long texts. Technical writers who use the page-oriented word processor may find it easier to edit printed copies. For more information about page-oriented word processors see Wheeler's chapter in this book.

In this chapter I have described how to analyze text content. Most writers would like to have an editor who offers only constructive criticism, who is always available, and who is tireless. That editor, if available, would probably serve us better than electronic analysis tools. However, electronic tools are better than the editor we don't have.

Spelling and grammar checkers really are simple minded—they check text against lists. They really don't understand our text. They make suggestions we needn't follow. A spelling checker may object to technical terms; a grammar checker may object to certain expressions. Whether these items need changing is up to each writer. That they are pointed out can be helpful to all writers.

My recommendation is to take full advantage of electronic tools without slavishly following their dictates. I have suggested ways of writing that assume rote electronic editing. These methods work for me; they may not work for you. Experimentation will help you find your own optimum procedure.

BIBLIOGRAPHY

Halpern, J. W. and S. Ligget. *Computers and Composing*, Southern Illinois University Press, Carbondale, Illinois, 1984.

Shostak, R. (ed.). *Computers in Composition Instruction*, International Council for Computers in Education, Eugene, Oregon, 1984.

Wresch, W. (ed.). *The Computer in Composition Instruction: A Writer's Tool*, NCTE, Urbana, Illinois, 1984.

CHAPTER 6

Beyond Word Processing: Computers in the Composition Process

FREDERICK M. O'HARA, JR.

A distinction needs to be drawn between dedicated word-processing systems and word-processing packages installed on general-purpose computer systems. In many cases, the difference is only in the eye of the beholder, for dedicated word-processing systems are frequently general-purpose computers that have been cosmetically and functionally altered to limit their capabilities and to make them simpler for their users. Such a general-purpose computer might have the labels on its keys changed from "F1, F2, . . . , F11" to "Insert Character, Insert Line, . . . , Delete Line" to make the functions of those keys in word processing more obvious to the user. And the boot-up routine of the computer might be changed to automatically load the word-processing software so the user does not have to be bothered with all those tedious descriptions of memory maps, BIOS locations, etc. and does not have to separately load the word-processing software package. As a result, the user is ready to create or edit copy almost as soon as the machine is turned on. The tradeoff here is that, to all appearances, word processing is all the user can do with the machine.

However, the heart of any word-processing system is a computer. Although the computer within a dedicated word-processing system can be lobotomized so that it cannot perform any other function, it usually is capable of performing a wide range of computational functions. For word processing, that computer simply runs a program that carries out the calculations and procedures of arranging text on a screen, storing that text in memory, and directing a printer to reproduce that text. The program that governs these word-processing functions is not unique in its ability to control the computer. All else being equal, the

computer will run any program designed for it, whether that program is a word-processing package, an accounting package, a stress-analysis program, or whatever.

As this fact became obvious to buyers and as the user community became more sophisticated, dedicated word-processing systems were displaced by small computer systems that came with word processing built into their repertoires. These systems require more user input to function, but in a day in which computers scan our grocery purchases and tally our bill at the supermarket checkout counter, few users (secretaries, students, writers, bureaucrats, or even scientists) should be intimidated by such demands.

Early on, I opted for a small business computer rather than for a word-processing device for a number of reasons. First, I thought my family would enjoy playing computer games; none of them did. Little time was available on the computer when I was not using it for writing, maintaining data bases, running invoices, or computing tax schedules. Second, I thought my children would benefit from learning to write computer programs. They did, but only under duress, usually the pressure of required schoolwork.

The system I put together is based on a small business computer with four serial ports, one parallel port, two megabytes of floppy-disk storage, and 196K of user memory. This system allows me to create, manipulate, and store text; to transfer that text to other devices (such as a client's computer in another city or to a printer); and to run a variety of other programs. To this I have added

- an editing terminal that allows me to give directions to the computer with a minimum of keystrokes and mental effort;
- a 1200-baud modem that allows me to communicate with a distant computer or phototypesetter, either sending information from my system to that remote device or capturing (downloading) information from that other machine;
- a line printer capable of printing 120 characters per second (cps); and
- a printer that types letter-quality output at 55 cps with superscripts, subscripts, multistrike boldface, and multiple typefaces.

On this system I run a word-processing package; a data-base-management system; an asynchronous communications package; several languages; and other applications programs, both commercial and self-developed. The use of this software is described below.

The relatively large memory (used to hold the information while it is being manipulated by the computer) and mass storage (used to store the information while the computer is turned off or performing another job) were needed for the tasks I wanted to undertake: the preparation of large manuals and book manuscripts. The first book I did on my system (O'Hara and Sicignano, 1985) took two 5¼″ diskettes (two megabytes) to store the text, front matter, appendixes, indexes, and backup copies of all the files. The second (O'Hara and O'Hara, 1984) took six diskettes (six megabytes) and included virtually all my notes and several versions of the text.

To this basic system, I have added a complementary system: a notebook computer. It has several benefits. I can take it many places I cannot take my basic system, such as to a library or on an interview. The notes I take on it, or the text I compose while traveling, can then be stored on magnetic tape or uploaded into my basic system or played out on a printer. With the telecommunications capabilities of the notebook computer, I can also use it as a terminal to control my basic system or to contact remote computers of clients through a modem. With this backup system, I can keep on working (at least on a limited basis) even when a disk drive goes down or a chip goes out on my terminal. These and other crippling disasters have occurred, usually at critical times in the production of a publication.

The use of a general-purpose computer opens the way to using a large number of other capabilities in addition to word processing in the production of documents. These serendipitous capabilities and the improvements in document quality that they allow are the main topics of this chapter.

COMPUTER CAPABILITIES AND WORD PROCESSING

Many of the applications of computer technology to the production of documents rely on the availability of hardware and/or software. And each package (be it hardware or software) usually has its own protocol or command language, which is almost never identical to or even similar to the user protocols or commands of any other package on the system. This "user unfriendliness" is a real barrier to use. Each time a new capability is tried, long hours must be invested in study, trial, and error. I have about forty-five manuals detailing how to use the software, operating systems, and peripherals of my system and how to use the mainframe computers that I connect my computer to. This difficulty of use notwithstanding, the writer who is willing to experiment, invest money, and spend time can develop a wide variety of capabilities to ease the production of documents with computer-aided composing. These capabilities fall into five categories:

- Utilities (Operating System Support)
- Communications
- Data Base
- Specialized Applications Programming
- Boilerplating

Operating System Utilities

The operating system of a computer gives the computer its fundamental instructions and interprets the commands given by the user into instructions that can be understood and executed by the computer. It is usually accompanied by a set of utility programs that are quite helpful to an on-line composer because

they provide additional capabilities above and beyond what a word-processing program might allow.

For example, a word-processing routine might have the capability of telling the user what files are on a particular disk or in a particular user area. It might even separate the listing into document files (for example, files with the extension .DOC on their names), backup files (previous versions with the extension .BAK), and "other" files. It might even sort the entries in each of these sections alphabetically so the user can find a particular file name more easily. The directory utility on a system will likely do many of these same tasks but also give the user additional information, such as:

- how many files exist on a disk,
- how many are allowed by the operating system,
- how many records have been used (an indication of the physical size of the stored material),
- how many bytes (characters) are stored on the disk, and
- how much space they take up on the disk (including the space left in each file for expansion during future revisions).

An example of such a sorted directory report is shown in Figure 2. With such information available, the user is less likely to complete a writing or editing job and find that he or she does not have enough disk space to store the created file and the backup copy of the original.

Utilities frequently allow the user to employ "wildcards" in commands, also. A wildcard is an asterisk or other symbol (depending on the system) used in a command to stand for any character or set of characters. Most word-processing routines will not accept wildcards in any of their commands, a fact that can make the copying of a large number of files and other tasks very tedious.

Wildcards can be used in the file specification used in a command, such as the common DIR (directory) command. A file specification typically is made up of a disk-drive identification (e.g., DSKA: or B:), a file name chosen by the user, and an extension. On large systems where a user is restricted to a single disk drive, the disk-drive identification can often be left understood (the computer "reads in" a default value). The disk-drive designation frequently ends with a colon. File names can be any unique character string the user chooses as long as it is not longer than the capabilities of the operating system (e.g., six or seven characters). The file name frequently ends with a period. The extension is usually limited to three characters, and users often use mnemonic extensions to identify types of files. (Many word-processing and other application programs automatically use or look for specific extensions, too.) Examples are:

- DOC for document files,
- BAK for backup or previous version files,
- BAS for Basic language files,

```
sdir b:
02:43:27 A:SDIR      .PRL

Directory For Drive B: User 0
```

Name		Bytes	Recs	Attributes	Name		Bytes	Recs	Attributes
ACE3SPEC	BAK	4k	7	Dir RW	ACE3SPEC	DOC	4k	8	Dir RW
BAYW1	BAK	32k	242	Dir RW	BAYW1	DOC	24k	190	Dir RW
BAYW2	BAK	32k	247	Dir RW	BAYW2	DOC	32k	246	Dir RW
BCH2	BAK	4k	28	Dir RW	BCH2	DOC	4k	28	Dir RW
BCH25	BAK	8k	47	Dir RW	BCH25	DOC	8k	47	Dir RW
BERKLIB	BAK	4k	18	Dir RW	BERKLIB	DOC	4k	19	Dir RW
BERKLIB2	BAK	4k	19	Dir RW	BERKLIB2	DOC	4k	6	Dir RW
BIOSKS	BAK	4k	11	Dir RW	BIOSKS	DOC	4k	11	Dir RW
BONPROP	BAK	4k	23	Dir RW	BONPROP	DOC	4k	26	Dir RW
BWEXTRA	DOC	8k	52	Dir RW	BWEXTRB	DOC	4k	13	Dir RW
BWFIG1	DOC	4k	1	Dir RW	EAST	BAK	4k	14	Dir RW
EAST	DOC	4k	14	Dir RW	ENT1	BAK	4k	6	Dir RW
ENT1	DOC	4k	6	Dir RW	ENT2	BAK	4k	4	Dir RW
ENT2	DOC	4k	4	Dir RW	ENTFORM	BAK	4k	4	Dir RW
ENTFORM	DOC	4k	4	Dir RW	HARVARD	BAK	12k	66	Dir RW
HARVARD	DOC	12k	71	Dir RW	HARVLET	BAK	4k	19	Dir RW
HARVLET	DOC	4k	22	Dir RW	LOH1215	BAK	16k	105	Dir RW

```
LOH1215   DOC    16k   105   Dir RW    MOH1112   BAK    8k    44    Dir RW
MOH1112   DOC     8k    44   Dir RW    MOH1210   BAK    8k    44    Dir RW
MOH1210   DOC     4k    20   Dir RW    MOH1212   DOC   16k   105    Dir RW
MPI       BAK     4k    12   Dir RW    MPI       DOC    4k    12    Dir RW
NEWBCH    DOC     4k    28   Dir RW    PCOC85    BAK   24k   183    Dir RW
PCOC85    DOC    28k   196   Dir RW    PCOCBIB   BAK    4k     9    Dir RW
PCOCBIB   DOC     4k     9   Dir RW    PEACH     BAK    4k    12    Dir RW
PEACH     ENV     4k     2   Dir RW    PEACH     LET    4k    12    Dir RW
PROV      BAK     4k     8   Dir RW    PROV      DOC    4k     8    Dir RW
PROVDNT   BAK     8k    45   Dir RW    PROVDNT   DOC    8k    45    Dir RW
SOLE      BAK     4k    31   Dir RW    SOLE      DOC    4k    18    Dir RW
UCSDLIB   BAK     4k    19   Dir RW    USCDLIB   DOC    4k    17    Dir RW
WAYWER    BAK     4k     5   Dir RW    WAYWER    DOC    4k     5    Dir RW
YCMIN     DOC     4k    13   Dir RW    YCMIN12   DOC    4k    32    Dir RW

Total Bytes     =   480k  Total Records =    2711    Files Found =    62
Total 1k Blocks =   367   Used/Max Dir Entries For Drive B:       62/ 177

0A>
0A>
```

Figure 2. A sorted directory report showing statistics on disk usage.

- DAT for data files, and
- COM for files containing programs that perform commands.

An extension, however, is not always necessary. It only needs to be used when several versions of a file have to be distinguished.

If I had written an essay on word usage and called my file WORDS, then I would probably have two files in my directory with that name: WORDS.DOC and WORDS.BAK, the document and backup files. To find out what versions of that file I have, then, I could use a wildcard in giving the command

```
DIR WORDS.*
```

The directory utility would then list all of the files with the name WORDS and *any* extension. If I wanted to see only what document files I had, I could give the command

```
DIR *.DOC
```

If I had put the different chapters of a book in different files named CHAP1, CHAP2, CHAP3, etc., and mixed them in with a large number of other files, I could list just the book chapters with the command

```
DIR CHAP*
```

Utilities also may allow wildcards to be used in copying or renaming groups of files. For example, under several operating systems a command like

```
PIP E:=B:WORDS.*
```

will copy all files on the disk drive designated B with the file name WORDS to the disk drive designated E, giving them the same names on drive E that they have on drive B. The PIP stands for Peripheral Interchange Program, a utility that is supported by many operating systems.

Other operating system supported utilities can show you:

- the amount of free space on a disk,
- the number of users on the system at a given time,
- the error-production history of a disk,
- the formatting specifications of a disk (how information is physically arranged on the disk, an important consideration if you want to use a diskette from one machine on another machine),
- the password protection of each file,
- the date of creation or last updating of a file,
- whether a particular file or drive can only be read from (a read-only or R/O file or drive) or can also be written to (a R/W file or drive), and
- what your user number and other identifying data are.

As a matter of fact, these utility programs themselves are sometimes used by other programs to carry out such tasks as copying or erasing a file. That is one

reason why word-processing software packages are restricted in use to a particular operating system (and thereby to a particular type of computer) or even to a particular *version* of an operating system. And it is often these very utilities that are used in a dedicated word-processing system to bypass the other capabilities of the computer within it. For example, with the operating system MP/M II, the command

```
$0$.SUP
PT
```

will automatically load the word-processing package when user number zero (**0**) turns on his or her machine. With such a command in place, all other capabilities and operations of the system become invisible and perhaps even unavailable to the user.

Communications

The most powerful capability that I have tapped is electronic document delivery. In its simplest form, I produce a manuscript on my system and then transmit that manuscript from my computer to the computer of my client over regular telephone lines or via satellite. The document is thus delivered to the client electronically. Some of the benefits of this capability are the speed of delivery, the avoidance of having to print hard copies off the computer, and the production on the client's own computer of a machine-readable version of the manuscript that can be run off through a printer or even a phototypesetter without re-keyboarding. When the transmitted file is to be output through a phototypesetter, the necessary system-specific control codes can be inserted in the file even before it is transmitted so the file can be run through the phototypesetter as received.

To transmit a document, I:

1. bring up the communications package on my computer,
2. dial up my client's computer,
3. turn on my modem,
4. log onto the remote computer,
5. copy the file from my computer to the client's computer,
6. log off the remote mainframe,
7. shut down the modem, and
8. exit from the communications package.

As a precaution against garbling of the file by line noise during transmission, I usually use an error checking option on my communications package. That option allows my computer to send a character, requests the receiving computer to echo what it has received, and then checks to see that what was sent is what was received before it sends the next character. It slows transmission significantly

(by a factor of three or four with my communications package), but it saves a lot of proofreading. With the error checking option in force, a 150-page book (with all its embedded commands for phototypesetting) takes about seven and a half hours to transmit at 1200 baud.

One might say that sending the text by Federal Express would be cheaper and almost as fast for distant delivery points. However, that argument does not take into consideration the variety of methods of transmission (and their associated costs) nor the technical complications of swapping media (e.g., diskettes or tapes) between one machine and another. Sending the copy in print form requires rekeyboarding with its attendant waste of time and money and its introduction of errors. Sending a machine-produced recording (tape or diskette) is fine as long as the machine and software of the sender is the same as that of the recipient. However, even with a technology that has been around as long as half-inch magnetic computer tapes, the variety of tape formats is formidable. The physical arrangement of data on the tape can vary dramatically from one computer system to the next. Even the representation of characters can vary drastically as standard (e.g., ASCII) or nonstandard (e.g., IBM's EBCDIC) codes are used. The differences among diskette formats is even greater. One has to determine whether the receiving computer uses 8-, 5.25-, or 3.5-inch; single- or double-sided; single-, double-, or quad-density; 48-, 96-, or 135-tracks per inch diskettes before determining the arrangement of the data on those diskettes. Knowing the brand of computer does not always help either, because different models put out by the same manufacturer might use different formats, or the owner of the machine might have customized it with larger-capacity drives and reconfigured the operating system accordingly.

Thus, electronic transmission of text files can become the great leveller. The transmission network will accept the text from whatever source you send it, and the remote computer will format the received information in the format that it always uses.

Transmission costs are largely a nonissue. A manuscript the size of this chapter would take less than five minutes to transmit without error checking. For that short a transmission, I would consider telephone transmission to almost anywhere an affordable alternative. For a seven and a half hour transmission, I would use the common telephone services only for the delivery of a manuscript to a local client. For longer distances, packet-switching networks are available and charge a flat rate of about $6.00/hr for transmission. Some of my clients have satellite earth stations, and I have used those facilities to transmit documents to them. The costs are not extraordinary because the system accumulates my data and then sends it through the satellite network very rapidly, cutting the transmission time significantly.

An example of how electronic transmission can be used to overcome the variations between computer systems is embodied in a book I coauthored and edited (O'Hara and Martin, in preparation). I wrote about 25 percent of it on my

computer, and the rest was written at various locations around the country. We wanted to save time and money by using the original versions prepared by the other authors. The problem was the incompatibility of the various systems used by the different authors. This problem was overcome by transmitting each of the contributions to a central host computer:

- The portion written at a large research laboratory was delivered on half-inch magnetic tape. It was read into one of my client's mainframe computers, and then copied off that mainframe into my computer over the telephone network.
- The portion written at another laboratory was delivered on floppy disks. A communicating terminal at another facility was hard-wired to the first mainframe, and the disk was read into the mainframe for downloading into my computer.
- The portion written at a third laboratory was also submitted on floppy disks. These were read into a compatible computer at yet another facility that could be networked to the original mainframe for downloading to my system.
- One of the contributions from the last lab was also delivered on disks. Again, these were read into the mainframe.
- The rest of the manuscript was delivered in printed copy that was run through an optical character recognition device into a computer and networked for downloading.

By these means, all these contributions from across the country could be integrated with the text written on my system even though they were prepared on a wide variety of incompatible systems. Virtually none of it was rekeyboarded, an important consideration with an 1100-page manuscript.

A capability closely related to electronic document delivery is the use of electronic mail systems. When I have a customer who has a mail utility on a mainframe computer, I frequently use that utility to communicate with the person in charge of the project. For example, while writing a manual for a computing group, I would log onto their computer through my system and would compose some of the text on their computer. This might be at 10:00 pm, 2:00 am, or 10:00 am. When I got to a point where I wanted to check if I was on the right track, I would quit and send an electronic mail message to the project coordinator. I would cite the file's name and its location on the computer and ask specific questions about the text. The next time the project coordinator logged onto the system, a message would be displayed: "YOU HAVE NEW MAIL." Answers to my questions could then be forwarded to me by the coordinator via telephone or a return mail message. Before computerization, such review procedures would take days. I would have to type up the manuscript, deliver it to the customer, and then wait for a chance for them to sit down and talk over their comments. With electronic mail, it is not uncommon for me to go through three review sessions with a project coordinator in a single working day. The result for customers is a product that is more

to their liking, and the result for me is less time wasted producing text that will ultimately have to be replaced.

Data Bases

Data base management systems (DBMS) have made a significant impact on composing text. They help the writer research the subject to be written about, they provide ready made text for inclusion in a document, and they allow manipulation and reordering of text stored on the computer. For example, I was writing a chapter of a book one day under a tight deadline (an expired deadline, to be more exact). I was describing research conducted on the storage of electric energy with superconducting magnets. The literature I was working with described an experiment conducted in Japan, but it did not describe the results. How could I bring the text to a logical conclusion without hours of library research to find the results of the work? I switched on my modem, dialed into a commercially available bibliographic data base, and "asked" if it had any citations to work published by the Japanese researcher. In seconds, I had a list of his publications and could identify the publication that held the research results. Another command to the DBMS brought an abstract of the paper onto the screen of my terminal. I had my computer make a copy of the citation and the abstract and logged off (total time, about three minutes). A telephone call identified a nearby library that held the cited journal, and I was able to get a photocopy and be back at my keyboard in less than an hour to finish the writeup.

In the case just described, the *information* contained in the data base was used in composing a document. In other cases, the actual *text* contained in the data base can be of aid in constructing a document. The data base can be searched, relevant data identified, the text downloaded into the local computer, and the downloaded file(s) edited with a word processor to produce finished text.

The parallelism between this procedure and the copying, cutting, and pasting of material from a printed book should be noticed. Data bases are frequently copyrighted just as more traditional publications are, and republication of that material can be an infringement of the copyright. Care should be exercised to ensure that infringement does not occur. Infringement will not occur if the publisher of the material (i.e., the client for whom the writing is being done) already holds the copyright to the material, if the material excerpted qualifies under the educational and scholarly use provisions of the Copyright Act (including what used to be referred to as the Fair Use Doctrine), if clearance is obtained (in writing) from the copyright holder, or if the textual expression of the information is substantially changed.

In other cases, a data base management system can be used to move text already stored on a computer. For example, I have used the DBMS on my computer to store the index terms and page numbers of subject and author indexes for a book and then had the DBMS sort the index entries, either alphabetically,

by volume number, or both. A list of 1,562 index terms was alphabetically sorted in ten minutes by this method. I shudder to think how long it would have taken me to sort 1,562 index cards manually and then retype the sorted list. Entries for directories and mailing lists can be handled in a similar manner with such a system.

Specialized Applications Programming

The availability of the capabilities of a computer system allow a variety of tasks to be automated and easily performed. When I first got my notebook computer, I wrote some programs simply to experiment with and learn the BASIC programming language that came with the computer. The first program was one that piped out "For Boston," the fight song of two of my alma maters. That programming task was trivial, as was the job that the program performed. But the purpose of the exercise was to teach me how to use the machine to do jobs that would otherwise be repetitive drudgery and/or that would require exactitude in performance.

Such a job is keeping track of the time spent by the various workers on our different projects and then producing monthly invoices to clients. For this task, I wrote a package of programs specifically designed for our family consulting business. The package is entered through an umbrella program that displays a menu that lists the capabilities of the package, asks what function is to be performed, and then selects the appropriate program to perform that task. The package is also exited through this umbrella program. Once in the package, the user can:

1. log in a project, essentially telling the computer
   - to whom the invoices should be addressed,
   - the contract identification number,
   - the total amount of the contract (so more than that will not be invoiced),
   - the amounts budgeted for labor, telecommunications, and travel (again, to place limits on the amounts that can be billed),
   - who is working under the contract, and
   - the hourly rate charged for each worker's labor;
2. log in timesheet data, telling the computer
   - the project on which work was performed,
   - the month in which the work was performed,
   - the date the work was performed,
   - the worker performing the work, and
   - the nature of the work;
3. prepare any of five types of invoice (request for progress payment, final invoice, etc) by telling the computer
   - what type of invoice is to be prepared,
   - what project is to be invoiced,

- what period is to be covered by the invoice, and
- to whom payment should be made, and

4. prepare a history of a project that lists the name of the project, the amount and date of the work performed, what work was done, and who did the work.

The program keeps track not only of the contract and timesheet data, but also the cumulative amounts spent for labor, telecommunications, and travel for each project. In addition, it performs all the calculations necessary to prepare the requisite invoices and prints them out according to a standardized set of formats.

One does not fully appreciate the convenience afforded by such a program until the hours of three workers having three rates of pay and working on four projects for four different clients have to be accounted for and billed on a monthly basis.

Other programs I have written and installed on my system prepare tax forms for those who provide consulting services to us, produce reports of travel expenses, and compile data on the business for use in filing tax information. Smaller programs sort, format, and display data for incorporation into documents.

Boilerplating

Boilerplating is the use in a document of text that was previously prepared for use in another document. For example, a manual on the repair of electrical equipment may have to alert the repairman of a hazardous condition occurring at several steps of the repair:

WARNING
Energized circuit.
Use insulated tools.

Instead of typing this warning in every time the condition is encountered in the manual, a writer can simply make up one such entry and copy it into each location it is called for.

The preparation of such a library of standard sentences, paragraphs, sections, or chapters for inclusion in a series of documents is a well established practice. In the days before computers, this replication was done with printed copies, scissors, and paste. Now, the use of computers to store, retrieve, insert, and update such boilerplate material has made the practice easier. No longer need the material be rekeyed or cut and pasted. Clean copy of assembled boilerplate can be produced with a few minutes at the terminal. And word processing allows the user to custom-tailor the contents of boilerplate to the document in which it is to be used. I have seen an instance where a company built a building that was exactly the same as another building they had built years before in another location. When they had to document the building plans for the new building, they simply copied the computer files for the old building's documentation and

changed the name of the building in the text files. The old document became the boilerplate for the new document, and the new document was produced at an extraordinarily low cost.

CONCLUSION

The point I hope I have made with this largely anecdotal essay is that computers offer writers and editors a powerful tool that transcends the capabilities of simple word processing. Computers, particularly locally operated microcomputers, allow writers and editors to assemble, manipulate, format, and print information much more easily, quickly, and sophisticatedly than was even imaginable twenty years ago. These computer capabilities allow the rapid transfer of information among machines and among widespread locations. They allow a wide flexibility in the method and appearance of output. And they spare the writer and editor from repetitive, clerical tasks so that that time can be devoted to more productive work. A computer system can perform these tasks not only through its word-processing capabilities but also through the capabilities of the wide range of hardware and software contained within it.

BIBLIOGRAPHY

O'Hara, Jr., F. M. and Robert Sicignano. *Handbook of U.S. Economic and Financial Indicators*, Greenwood Press, Westport, Connecticut, 1985.

O'Hara, Jr., F. M. and Laura O'Hara. *Sourcebook: Gaining Access to U.S. Government Information on the Environment and Natural Resources*, U.S. Department of Energy, Washington, D.C., 1984.

O'Hara, Jr., F. M. and J. F. Martin. *Recent Research on Energy Storage Technology*, in preparation.

PART III

Graphics and Electronic Publishing

GRAPHICS AND ELECTRONIC PUBLISHING

Basically, word processing allows easy and efficient text entry. Coupled with this function is the ability to make corrections and revisions and to format texts. These texts can then be stored electronically for subsequent printing or revision. Recent innovations have added elaborate graphics, efficient telecommunication capabilities, and integrated electronic publishing to these basic functions. With these new capabilities the roles of writer, editor, graphic artist, layout designer, and production manager have begun to merge into one performed by the writer alone.

BASIC GRAPHIC FUNCTIONS

Even the most rudimentary word processors provide, at least, the ability to create tables. Tab functions, often coupled with a limited set of graphic characters, allow the writer to prepare tables with only a few keystrokes. Slightly more sophisticated programs allow writers to place graphics created with other computer tools, such as spreadsheets or accounting programs, into word processing documents. In such programs, provisions have been made to create electronic notepads or clipboards that move electronic art from one application to another.

"Click" Art

Moving beyond these basic graphics requires programs that allow the writer to use the "graphics pages" available on their system. Usually there are high and low resolution graphics screens. To use these areas the writer must either have

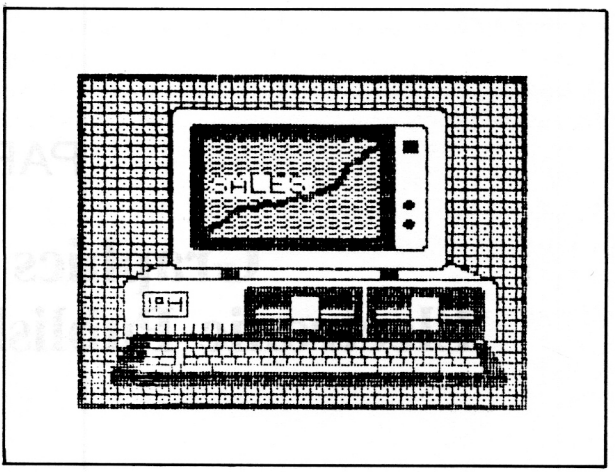


Figure 3. "Click-Art" graphic.

access to preprepared (canned) art, usually called "click art," after its older paper form—"clip art"—and/or some kind of graphics input device.

The simplest method for the nonartist who wants to include artwork is to find a system or program that provides canned electronic art. Such a program provides several ways to interact with the artwork. The most common method, a menu-driven program, allows browsing through a directory of stored electronic art. Specific "pieces" of art can be selected and brought into a document on a clipboard similar to that used for incorporating charts and graphs. This art can then be positioned in a document.

Versatile click art programs allow the user further control by providing a means of editing art. For instance, a user can select a visual of a piece of office equipment that is similar to the one she markets. That visual can be "edited" by adding fill patterns, changing basic shapes, or adding identifying letters. She can even zoom in on specific areas to edit on the pixel level. Figure 3 provides an example of such artistic efforts.

Electronic Fine Arts

While "click" art type programs allow writers to use many of the tools of a graphic artist, they cannot provide the same range of graphic abilities that a good graphic artist possesses. At the most sophisticated level, electronics can provide a full range of graphic techniques similar to those used to create paper-based art. At the most basic level, these systems provide a method for digitizing art—converting images from paper to electronic signals. A host of graphic input devices offer this capability.

page not scanned by accident

She offers advice on how to use rudimentary word processing functions to create highly *useable* page designs in Computer Graphics for the Technical Communicator. William Benzon gives us a glimpse of the graphic capabilities one can expect to be able to perform on basic word processors with some digitizing and manipulation capabilities. He demonstrates how images are brought into the electronic environment and how they can be selectively altered before they are incorporated into a text. Together these pieces describe the range of available electronic art. These capabilities have considerable interest and importance for producers of technical documentation, especially given the highly graphic nature of these texts. David Orr provides an overview of the ways in which text are prepared and edited for typesetting and telecommunications in Text Preparation and Transmission for Word Processing. Finally, in Word Processing and Electronic Publishing, Philip Rubens offers a preliminary glimpse at the kinds of advantages and problems inherent in desktop publishing.

CHAPTER 7

Using a Word Processor for Page Design

PATRICIA CAERNARVEN-SMITH

By now you know how to use a word processor as a technical communication tool to capture keystrokes and edit text. Many people believe this is enough: they produce pages and pages of text and call it technical communication. But you can go further; using your word processing system *as a design tool* to develop interesting, readable pages. This chapter shows you how.

Regardless of the size of your word processing system, you can perform some or all of the design tasks described in this chapter. To understand how much you can accomplish, you should understand two issues: page designs that work in technical communication, and the capabilities of your system.

WORD PROCESSORS AND PAGE DESIGN

Technical communication uses three methods of putting words on pages: typing, typesetting, and word processing. Typing is the oldest technical publishing technique. It has been used since the early 1940s to produce technical reports, simple manuals, and most technical drafts headed to technical publishers. The typewriter is potentially a highly flexible *design* system for technical publishing. It can justify, produce bold face and proportional type, install centered or other formatted text and math for display, and its output can be corrected for typing errors, proofreader's comments, and author's alterations.

However, most users limit their efforts to basic production: they begin typing in the upper left-hand corner of the page, and fill the page with words. The number of words on the page and the shapes of the paragraphs are limited by the overall size of the paper, the margins at top, bottom, and sides, the number of carriage returns required between lines and between paragraphs, and the typist's imagination.

Typesetting is an even older publishing technique, but was rarely used in the early days of technical publishing (except for textbooks) because of its expense. Today, an expert designer can make typesetting a highly effective tool in technical communication. However, typesetting is rarely used to its full effect, and some of typesetting's most useful attributes can be emulated successfully on some word processing systems.

Word processing lives between typing and typesetting. It uses a keyboard and terminology from typing: "carriage returns" and "lines-per-inch" rather than point sizes and leading. Word processing is computerized like typesetting and places some responsibility for format and processing in the computer. Originally designed as a method of automating the typing process, word processing can also be used successfully as the "front end" for automated page design and typesetting systems.

Some word processors turn out pages that are almost as precise, readable, and attractive as typeset pages. Here, we are concerned with designing pages on the word processor, to get an effect that is better than typing without the expense of typesetting.

THE PAGE DEFINED

A page is a sheet of paper containing a body of information. A page is limited by its overall size, its binding, and its margins. The "image area" of the page is the area in which the information is presented. Note that running headers, footers, and page numbers typically occupy top and bottom margins of the page outside the image area as shown in Figure 4.

The information in the image area may be a free-flowing presentation or a modular one. In a free-flowing page, the information fills the image area. The page may begin with the end of a hyphenated word continued from the previous page and end with another. The result is called a "flush bottom" because the bottoms of the pages are flush with one another. This book uses free-flowing pages. Only the ends of the chapters stop short of the bottom of the page, and the only headings that begin a new page are the chapter heads.

A modular page contains a single message, often set off by a heading at the top and white space at the bottom where the information ends. In a modular page, the information may or may not fill the image area because the information ends when the concept ends. This presentation is called "ragged bottom" and is used in some of the best computer manuals. Such a format is shown in Figure 5.

Free-flowing or modular, the information on the page includes words, graphics, illustrations, and tables. Graphics include rulings, arrows, company logos, and symbols. Illustrations include photographs and drawings.

The words include paragraphs, lists, and tables. Paragraphs may be continuous, or separated by space. The paragraphs may begin flush left, or have their first

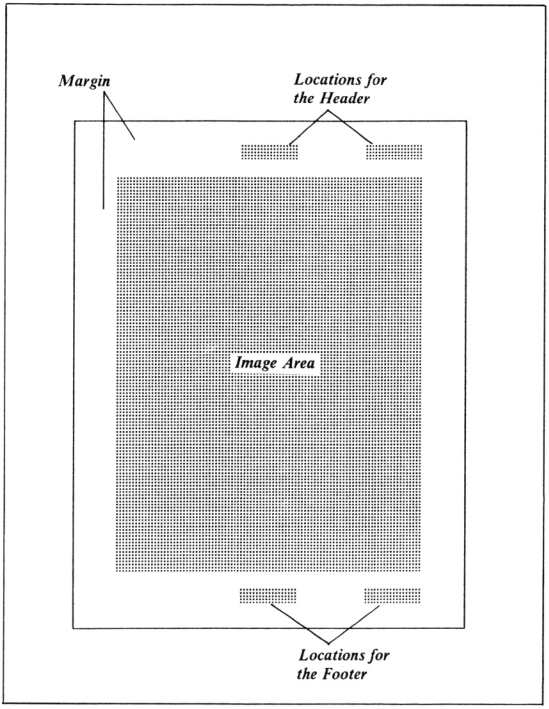


Figure 4.

lines set in from an identifiable left margin. Lists may be simple or numbered, indented, or bulleted. Tables may be ruled or unruled. In addition, headings and subheadings are used to introduce new concepts and separate blocks of words. Many of these elements are shown in Figure 6.

These page elements contribute to a reader's ability to scan the page for important information, read paragraphs and lists, interpret illustrations, and gather the information successfully.

PAGE STYLE DEFINED

Technical communication has developed some standard page styles that combine these elements:

- The full-page width, free-flowing style with imbedded headings. This page may be single-, dual-, or even triple-column.
- The super-wide page, typically single-spaced, free-flowing style with imbedded headings.

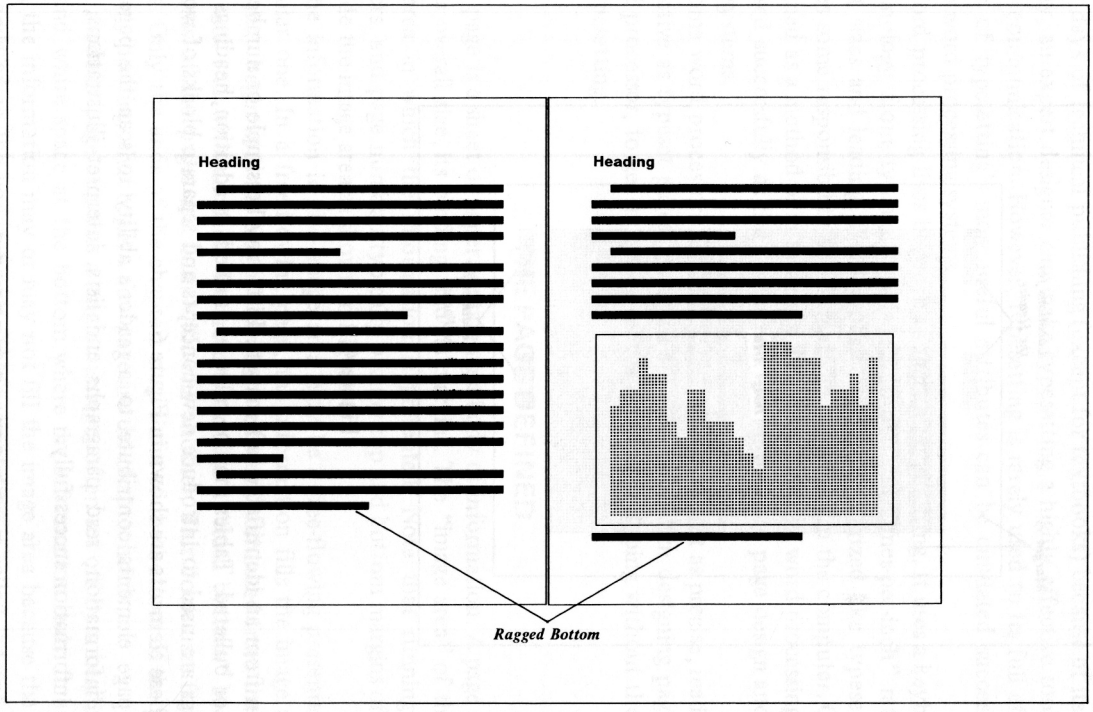


Figure 5.

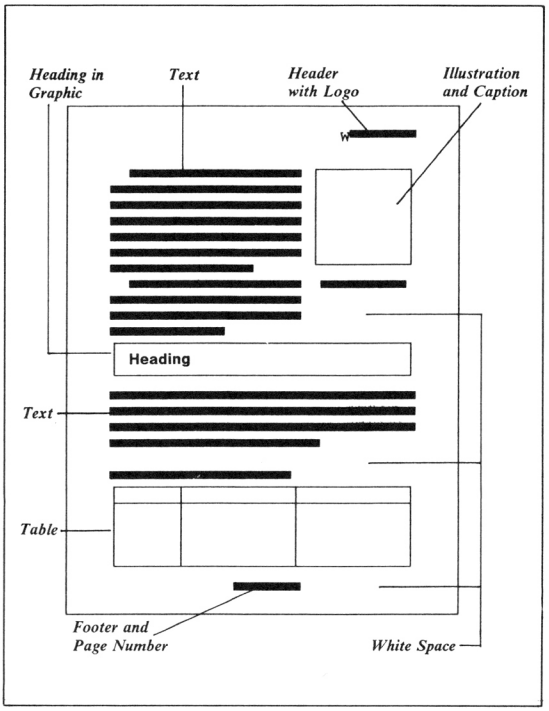


Figure 6.

- The reduced column, free-flowing or modular page, with top-of column, imbedded, or set into the margin.
- The dual-column, free-flowing or modular page, with top-of column or imbedded headings.

Full-Page Width

The full-page width, free-flowing style with imbedded headings is based on typing and traditional typesetting. The result is a page very full of information. The image area (on a sheet of 8½- X 11-inch paper) is typically eighty to ninety characters wide and twenty-four to sixty lines long (depending on whether the text is double-spaced). The text is centered left to right, top to bottom with approximately equal margins all around.

The high number of words accounts for the fact that the lines of text must be separated in some way by lines of white space. When a reader attempts to read a column more than fifty-five to sixty characters wide, his eyes have trouble staying

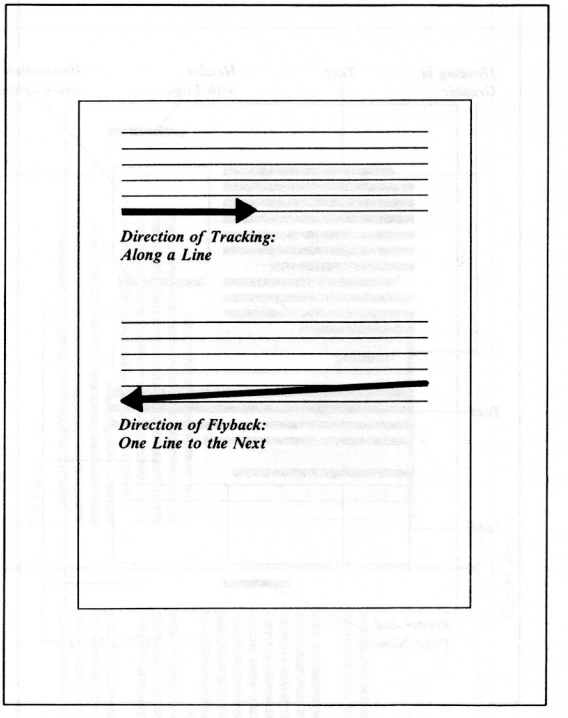


Figure 7.

on the correct line (tracking) and returning (flying back) to the next line (Figure 7). Wide columns are particularly difficult to read. Double-spacing between lines and/or between paragraphs is a compromise between putting a great many words on the page and helping a person read it.

In full-page width books, new chapters typically begin on new pages with headings centered near or at the top of a new page. Other headings are usually imbedded in the text, set off by one or more carriage returns, and may appear near or at the bottom of a page.

This page is not very flexible. Illustrations and tables often appear on separate pages from the text because of the problems of integration. Nevertheless, millions of pages are typed, word processed, and typeset in this style each year simply because "that's the way we've always done it."

Super-Wide Page

The super-wide, single-spaced, free-flowing page is based on the use of reduction techniques that shrink a large, typed, word-processed, or typeset page to a

smaller size in the printing process. The image area of the page may be as many as 100-145 characters wide, and the page length may be as high as eighty lines. At the same time, the page size may be reduced—sometimes to 5½ X 8½ inches or 4 X 6 inches. These pages are extremely difficult to read because of the problems of distinguishing words, tracking, and flyback. Typically these pages are created on 8½- X 11-inch paper, or on larger sheets, such as 11 X 14 inches, and then reduced.

In this format, illustrations and tables can often be placed quite artfully. For example, an illustration that requires less than the full horizontal page width may be accommodated by narrowing the column (fewer characters per line) for the length of the illustration. Below the illustration, the column returns to full width.

The argument for using these pages is that they reduce the number of sheets of paper required to present information. In the late 1970s, this was an argument in favor of "saving a tree" in a blizzard of increasing paperwork. More recently, this same argument is being used to reduce the overall thickness of a book (from, for example, a 3-inch ring-binder to a 2-inch binder) to encourage the publication's use. When these arguments are used, it is not surprising that the text is free-flowing and the headings are imbedded in the text. Nor is it surprising that these publications are not well used.

Reduced Column Page

The reduced-column page is a significant departure from the two previous styles. This style accomplishes several things. It accommodates the reader's eyes, because tracking and flyback problems are reduced. It also makes a more attractive page, by allowing some variety to the layout and by adding white space to the page.

The column is typically fifty-five to sixty-five characters wide. On an 8½- X 11-inch page, this column allows significant white space in the left or right margin (often used by the reader to add his own notes). The column can be any vertical length within the reasonable printing and trimming length of the page. However, designers who limit their column width also tend to limit its length as well. Thus, pages are more likely to be modular, with a heading at the top and white space at the bottom (ragged bottom).

These pages allow for quite flexible placement of illustrations and tables. Large illustrations and tables can fill the page or the lower regions of the page to its full width. Smaller ones fit into the column with text, and even smaller ones may reside in the wide margin of the page.

When the reduced column page is modular, major headings often appear at the tops of the pages. In the free-flowing versions subheadings may appear anywhere in the vertical column. Of importance here is horizontal location of the heading compared to the left and right margins of the text (Figure 8). With reduced column width, headings may be:

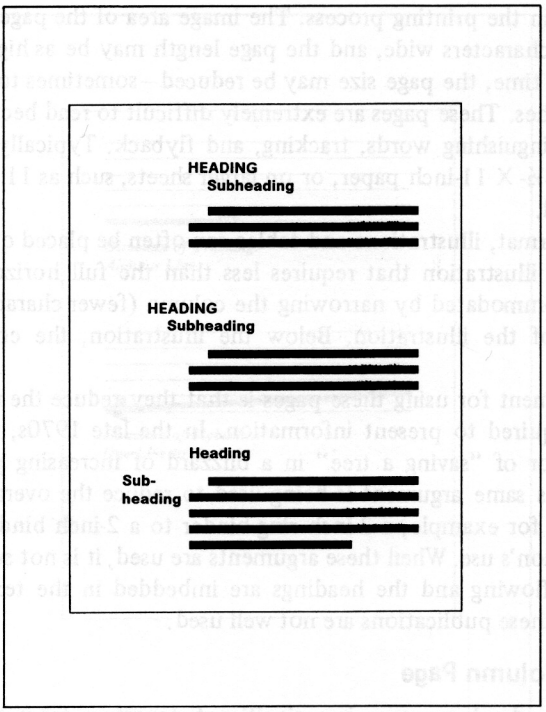


Figure 8.

- flush left with the column,
- tabbed to the right of the left-hand margin,
- set to the left of the left-hand margin, or
- set into a wider left margin and flush with the first paragraph "under" the heading.

Dual Column Page

The dual-column page, modular or free-flowing, with imbedded or top-of-column headings, is a style long favored by magazines and some branches of the military for manuals. Even when designed oversized and then reduced, the dual-column format limits the column width of the text to a reasonable line (typically 40 to 60 characters) while maximizing the number of words on the page and allowing for quite flexible sizing and placement of tables and figures. (See Figure 9.)

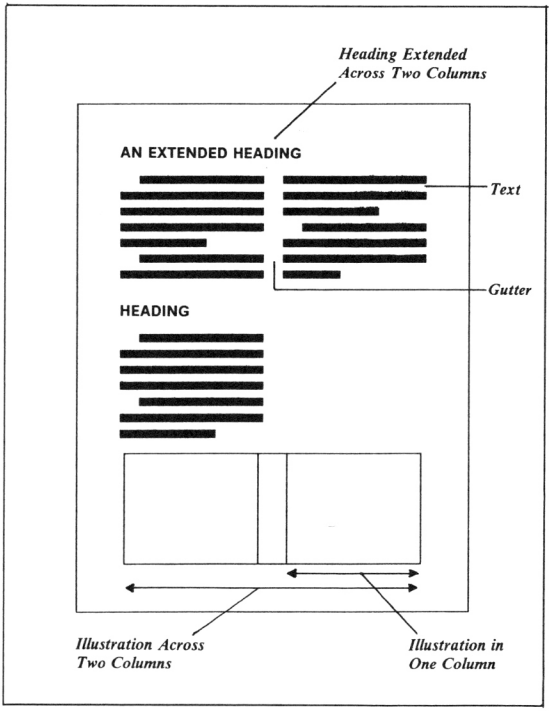


Figure 9.

Dual-column presentations are always single-spaced within paragraphs and often single-spaced between paragraphs (with tabbed-in first lines). Headings are placed at the top of the column or imbedded in the text as they occur. Tables and illustrations are placed within a single column or across both columns.

THEMATIC VARIATIONS

Within all these styles many variations are possible. Simply by changing the spacing between paragraphs and between lines, tabbing the opening line of each paragraph, or changing the column width by as few as two or three characters, you can begin to develop a distinctive and more readable style. As we shall see, much of what you can accomplish has to do with the capabilities of your word processor, and these capabilities will be explored here.

SYSTEM CAPABILITIES

Using a Word Processor to Design Pages

Your word processing system is a tool for designing readable technical publishing pages. To understand its use as a design tool, we must discuss some terms that pertain to word processing. Table 1 presents a glossary of more or less standard terms used in word processing. Note that many have their origins in typing. Others have their origins in the fact that a word processor is, after all, a computer. Thus, we must talk about file structures, ASCII files, and line feeds when we talk about word processing as well as talking about carriage returns, tabs, and indents.

Table 1. Word Processing Terms Used in Page Design

Term	*Description*
absolute left margin	The left-most position in which a character can be placed on the screen or printout can be manipulated on most printouts.
ASCII file	ASCII = American Standard Code for Information Interchange. In an ASCII file, characters are represented by their ASCII equivalents. Many word processing packages can create an ASCII file and can read one created by another word processing package.
carriage return	A formatting symbol inserted by the user into the text, indicating the end of a line or paragraph, and advancing the display or printout to the next line of text.
column width/line length	The total number of characters per line of text allowed 1) by the system, and 2) by the format code within the system.
decimal tab	A character or function provided by most bookkeeping packages and some word processing packages to allow alignment of characters *flush right* along the decimal tab column.
fixed pitch	A system of printing in which each character occupies a fixed fraction of an inch of horizontal space; for example, Pica type allows ten characters per inch.
flush left	Text that is aligned along a left margin.
flush right	Text that is aligned along a right margin.

Table 1. (Cont'd)

Term	*Description*
forced page break	The provision in word processing to end a page at the user's discretion. Some systems require fixed software to be overridden, hence the term "forced."
forced white space	The provision in word processing to insert white space of a certain shape into the text. Some systems require extensive effort for this.
format	A coding method that establishes the positions of margins, tabs, and indents for a text file.
format file	A separate code file, required by some word processing systems, for use with a text file.
format line	A line on the screen that displays the location of tabs and other formatting information. This information appears as either symbolic or numeric indicators.
indent	Alignment of a left margin against a position *to the right* of the actual left margin. Used for lists.
justified	A method of aligning the text along a margin; adjustments to the spaces between words and between letters provide the necessary alignment.
left margin	The location of the first non-indented non-tabbed character on the screen or in the printout.
line feed	A (typically) invisible character at the end of each line used to advance the cursor or printhead.
lines per inch	A number of lines of text per vertical inch allowed by a printer.
page file count	The total number of pages allowed in a file under word-processing software control.
page length	In word processing, the number of lines of text allowed on the printed page or in the display.
page width	In word processing, the number of characters allowed per line of text.
print routines	Software utilities in a word-processing package by which a word-processing package sends a text file to a printer, and by which the printer produces the page.
proportional space	A process of compressing or expanding the space allotted to each printed character to accommodate its width relative to other characters. For

Table 1. (Cont'd)

Term	*Description*
[proportional space]	example, the printed line "MMMMMMM" would, in proportional space, be allotted more horizontal width than in fixed pitch, while the printed line "iiiiii" would be allowed less horizontal width. Used in word processing when available to provide a more professional, more nearly typeset quality.
reformat	The act of changing text from one format to another, by changing the number of characters allowed per line and/or by changing the positions of tabs and indents.
tab	A horizontal position to the right of the left margin against which a tabbed paragraph or a column of a table can be aligned.
unjustified	"Ragged right;" printing that is not clearly aligned along a right margin.
wrap	The ability of the display or print routine to complete a line of text and begin the next line without input from the user.
WYSIWYG	*What You See Is What You Get*, pronounced wizzywig. The ability of a software package to display on the screen the exact results that will appear on the printed page.

Learning to use your word-processing system as a page-design system necessarily involves you in some exercises that ordinarily typists and computer operators never perform. To them, it is enough to capture keystrokes, edit for grammar, check the spelling, and turn out the pages. You are choosing to go beyond these ordinary applications because the extraordinary power that word processing provides to create and manipulate words also gives you the power to manipulate the form that those words take.

However, each word-processing package handles features in a different way. For example, some packages maintain formats used in publications as files separate from your text files; others as part of the text files. Some display the format line at the top of each screenload of text; others at the top of the page, or only at the top of the file. Some allow several or unlimited formats per document. Others allow only one. Some allow you to reformat the entire document with a few simple keystrokes; others require that you reformat every paragraph

independently, and that you remove line feed characters *at the end of each line* during the reformatting. Some packages allow you to use function keys to define indents, tabs, and line spacing. Others require the use of commands imbedded in the text. To design pages successfully, you must learn *how your system provides the features you need.*

When you begin to design pages, you must learn how your software performs to understand the ease or difficulty of such design steps as:

- formatting simple text in a readable page;
- adding significant material to an already established page and file;
- placing subheadings in the margin;
- changing the format of a table to accommodate a change in the overall page;
- creating a dual-column format;
- converting a text in fixed pitch to the same text in proportional spaced type; and
- adding running headers and footers.

Formatting Simple Text into Readable Pages

To begin, let us say that you have been given a technical report written by an engineer and typed by a start-at-the-upper-left-hand-corner-and-keep-going-until-you-run-out-of-page typist. This report is in a single ASCII file. The text is eighty characters wide, double-spaced. Page breaks have been inserted arbitrarily every thirty lines of text (59-line) page. The paragraphs begin flush left and are separated from each other by three carriage returns (one at the end of the paragraph, plus two more). The headings, including the main chapter headings, are imbedded in the text. The text contains two long lists that can be made into tables, and a page of captions for illustrations. (See Figure 10.)

You would like to re-format this text into a report with a fifty-five-character text column width, to be printed single-spaced on 8½- × 11-inch paper for review. You know that two reviewers will write copious notes in their copies, while an editor uses a third copy to turn this report into a slide show. Thus, all need plenty of white space.

Each heading down to the third level will begin a new page. Each paragraph will begin with a tab, will end with a carriage return, and will be separated from the next paragraph by an additional carriage return (blank line). Each list will be set off by bullets aligned on a tab that is set five characters in from the left margin. Each page will include room to integrate illustrations and their captions into the page.

Note: This is a big job, calling for patience. The procedure is as follows.

1. Convert the original ASCII file into one or more files suitable for your word-processing package. To do this you will have to control:
   - *An ASCII "read" procedure for your word processor:* Your word-processing software must read the original ASCII file and convert it into one or more

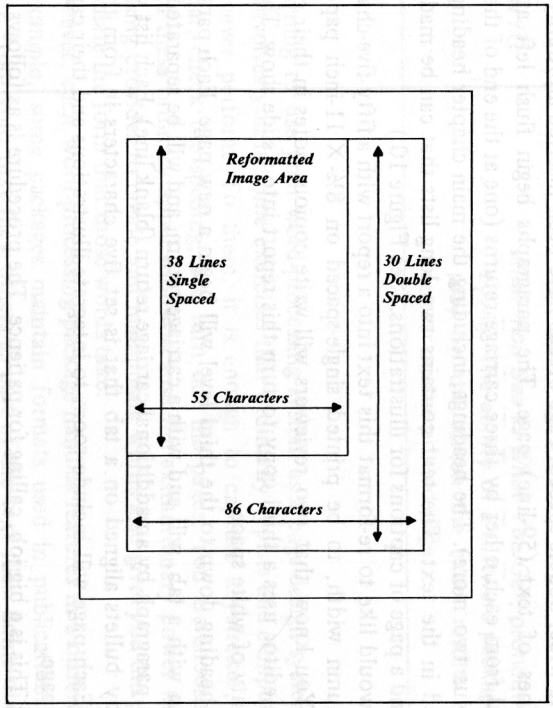


Figure 10.

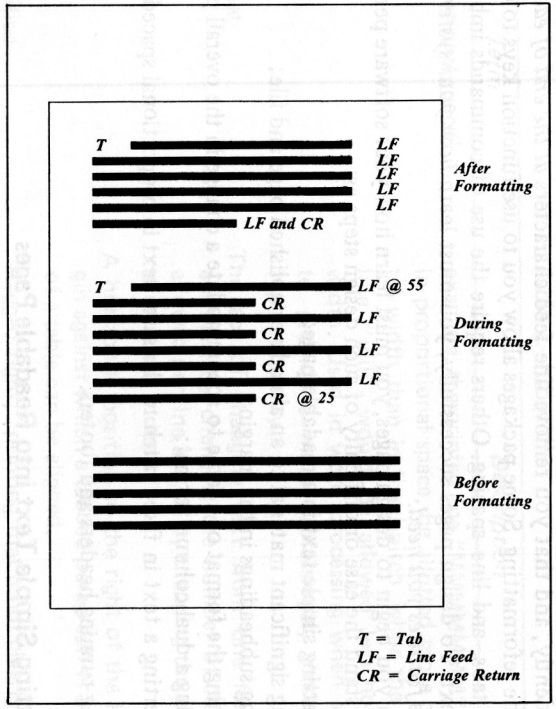


Figure 11.

new files that it can use. This process takes time, and may produce odd graphic symbols and strange character strings that you must delete. It may also delete tabs and other characters.

- *File structure in your word processor:* Whenever you begin significant format work on a document, it is best to divide that document into a number of consecutive or linked files *and back those files up* (copy them to diskette). One way is to create a file for each chapter, divide your original ASCII file into chapters, and move each into its new file. In this way, you can minimize any problems you might have with restrictive file page count limitations in your software's file structure.

2. Convert the eighty-character text column width to fifty-five characters:
   - *The format:* Reformat the entire document (one file at a time) to change the column width. If your system maintains formats in separate format files, you must create a file with the format you want (fifty-five characters wide with at least two tabs), then attach it to the file on which you are working. If your system maintains formats within text files, copy the format from one file to another rather than recreating it. Copying reduces errors and variations from chapter to chapter, file to file.
   - *Carriage returns and line feeds:* If the original file included a line feed at the end of each line, these may remain after the ASCII files are converted and the pages are reformatted. If so, each eighty-character original line is replaced by a fifty-five-character line and a twenty-five-character line. Delete these line feeds. See Figure 11.
   - *The page length:* Inevitably, decreasing the column width means increasing the page length for the same number of characters. In our example, the original document was eighty characters per line X thirty lines of text, a maximum of 2400 characters. Converting this to fifty-five characters per line can create a page of forty-four lines. You may want to adjust the lengths of your pages.
   - *WYSIWYG:* Finally, you want to see what you have accomplished. In many software packages, your screen display gives you an approximate view of the page. If not, now is the time to print the report for the first time to see what you have accomplished.
3. Begin each paragraph with a tab; replace double carriage returns with single ones. To do this, you will have to control:
   - *The carriage return and line feed:* If you removed extraneous line feed symbols in step 2, your pages probably still contain three carriage returns after each paragraph. They may *still* contain line feeds that are not needed by the new format. Delete any extraneous ones.
   - *Global search and replace:* Your old paragraphs close with three returns: one to close the paragraph, and two to separate paragraphs. Each new paragraph will end with a return and a tab: a return to close the paragraph, and a tab to open the next one. If your system has the capability, perform

a global search and replace procedure to replace one combination with the other. Note that removing extra returns between paragraphs shortens your page.

4. Place all first, second, and third level headings and their associated text on new pages. You must control:
   - *The forced page break:* Some systems allow you to install a page break anywhere in the text, even on otherwise empty pages. Others make this difficult, or impose a penalty (in file structure, print routines, or other problems). Simple or complex, the need to place each heading on a new page means you will be installing page breaks *just before each heading*. You will also remove the arbitrary ones used by the last typist.
   - *The file page count:* Remember the limits your software places on the number of pages in a file. You may have to create some more files.
5. Integrate illustrations and captions into the text. To do this, you must control:
   - *Forced white space:* If you have the illustrations for the text, now is the time to measure each to determine whether 1) it can be used in its present size, or 2) it must be enlarged or reduced to fit a possible space in text. Assuming that the final size of each illustration is *less than the total size of the image area*, your problem is to determine the *number of lines of white space* that must be inserted into the page that will hold the illustration.
   - *Lines per inch:* To do this, assume that you are printing out the result at six lines of text per inch of page. Measure the vertical size of the illustration *in inches*. Multiply the result by six lines per inch. The resulting *number of lines* is the amount of white space that must be inserted. You can install white space by inserting returns at the end of the appropriate paragraph or by entering appropriate commands, or by installing page breaks, depending on your system.
   - *Moving captions:* Now, move the appropriate caption from the caption list to the page, either above or below the white space allotted for the figure. Now, check your page lengths again.
6. Bullet the lists. To do this, you must control:
   - *The indent function:* Bulletted lists are by their nature aligned along some left margin. Since a list may contain single items that are longer than the main column, you should use *indent rather than tab* to ensure that a single bulletted item "wraps" for as many lines as needed.
   - *Simple graphics:* A bullet need be no more than a lower-case "o" that you fill in with a pen after the printout is complete. However, if your word-processing software gives you boxes, bullets, and other graphics, you can use them instead. See Figure 12.

Adding More Material to Existing Pages

Let us continue by supposing that your reviewers added significant material to several pages of this report. They liked the format, and wish to have the final

report—to senior management and selected board members—presented in the format you established. One reviewer has suggested the conversion of several paragraphs into bulletted lists to help senior management read the lists. In addition, both legal and marketing have added new chapters.

1. Assess the damage. Your new expertise in design will alert you to several problems: the need for new files to accommodate the extra chapters; an eye to page count limitations by the file structure; and the need to reformat paragraphs into lists.

2. Create new files to contain the two new chapters. If your files are numbered, you can create a complete new file structure to maintain the sequential numbering. If your files are named, make sure that the names of the new files contain clues that link them to the existing files.

3. Format and keyboard the new files. To do this, you must copy the format of your files or attach your format file to your new text files.

4. Make the rest of your editing changes.

Placing Subheads in Margins

Let's go one step further. Suppose the final report is presented to management in such a way that the impatient executive can scan down each page and read only the sections in which he is interested. Each heading down to the third level already begins a new page. But the report also contains a significant number of lesser headings. To facilitate skimming, you suggest placing these subordinate headings in the left margins ("glossing").

1. Change the format of each file to one that allows for locating these heads in the left margin. Instead of a format that is fifty-five characters wide with at least two tabs, you now need a format that is at least eighty characters wide. Your inset heads will rest against the absolute left margin of the page. See Figure 12.

Your text will begin at least twenty-five characters to the right of this left margin, and will require at least three tabs: one for the left indent lines of each paragraph (that is, the left margin of the paragraphs), one for the tab at the beginning of each paragraph, and another to locate the beginning of text for each bulletted item. (See Figure 13). In addition, you may want an extra tab to be located five characters to the left of the main paragraph tab: use this tab to locate the beginning of the main head at the top of the page to make it more readable. Globally reformat all of the chapters/files.

2. Globally replace the return-tab combination separating paragraphs with return-indent-tab. The return completes the first paragraph. The indent establishes the left margin of the paragraph (creating room for the inset heads). The tab moves the beginning of the paragraph in from this new left margin. You have now, in effect, moved the column of text to the right by twenty-five characters to make room for the inset heads. (Note that if any paragraph still begins at the

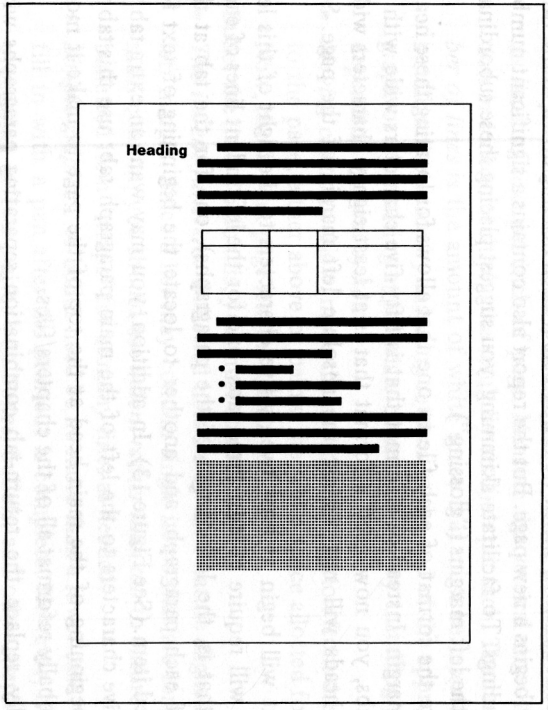


Figure 12.

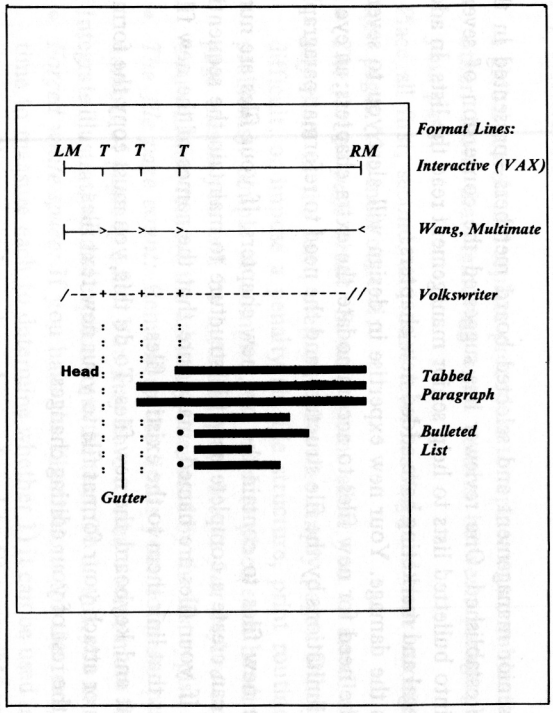


Figure 13.

top of a page, you must move it manually to align with the rest of the paragraphs.)

3. Insert each inset head into the new left margin. Each single-line inset head of twenty-three characters or less (allowing at least two spaces for a small gutter) can be inserted in the same way that you insert other text. Each inset head of more than twenty-three characters must be broken into two lines, and you can handle these in one of two ways:

- Leading line above the paragraph
- Flush with the paragraph.

Changing Table Formats for Page Fit

Of all the technical communication created on a word processor, tables are by far the most demanding. Tables require rigorous and iterative use of formats.

To be successful as a *table* processor, a word-processing system must allow direct, on-screen access to the format line, and the ability to use more than one format in a single file. You need direct access to the format line to allow for minor adjustments in horizontal spacing that allow for rulings and readability between columns. You need more than one format per document in order to develop a table as a continuing part of the page, integrated with the text, and available for editing. If you have neither of these features, you should plan to create a separate file for each table. When you print out all of the files, you must cut out the tables and paste them into the text.

1. Examine the table in its original form. How many characters wide is it? How many lines long is it? How must it be changed to conform to your new format? Very often, the most important work on a table is in rewording some of the entries, using abbreviations and acronyms, numerals instead of spelled-out numbers, and symbols instead of words and numbers. Pay particular attention to the headings of columns. A heading may contain more characters than the information within the column. Cutting down the heading can cut a table down to size.

2. Examine the format line that controls the table. Change the format 1) to fit the overall width of the page (in characters), and 2) to accommodate your revised entries.

3. Apply the new format to the table, and continue to edit. (**Note:** use tabs, not indents, and be careful that entries aligned horizontally do not touch or overrun each other. Until you are very good at formatting tables, print them in fixed pitch type.)

4. Print out the table.

Creating Dual Column Documents

Most systems make dual-column printouts difficult or impossible. Others make dual column quite easy. Some even display both columns on the screen

while you work, but force a penalty by taking away your ability to control the bottom of each column.

Even when a system allows creation of a dual-column document, the process may be involved. For example, your system may require left-hand columns in one file and right-hand columns in another—and may then print them as a merged-print document. Thus, your ability to create a dual-column document may demand more of your ingenuity than it demands of your word-processing system.

1. Find out if your system makes any provision for printing two simultaneous columns on the same page. In most systems this is through the print routine, not the editing process. To do this, you must understand control of:

- *The print routine for your system:* If your system provides for dual-column printing, you must learn how to control the print routine. Learning the print routine and setting it up may be so complex that you would do better to print out single columns and paste-up the results into dual columns.
- *Formatting for dual columns:* Each column of a dual-column document is typically forty to sixty characters wide. Dual-column pages also require a three to ten character margin ("gutter") to allow for readability. If, however, either your software or your printer sets a limit on the number of characters *in the printout width*, you must calculate carefully to keep the print routine from aborting. With a limit of eighty characters, for example, you can print two columns of thirty-eight characters with a four-character gutter, and stay within your eighty-character limit. Change a format to thirty-nine or forty characters and your printout will abort.

2. Set your format accordingly. Simple arithmetic, such as we used in the previous paragraph, suggests that the format for each column must be kept simple. If your column width is maintained at forty to sixty characters, and you begin to indent extensively to set off, say, a procedure with sub-steps and branching decisions, your column can become very narrow indeed. Formats have already been discussed, and by now you should have a good working knowledge of their control. You must use the same format in each column, or the reader will see a lop-sided and confusing page. Figure 14 shows the format characteristics of the dual-column page.

3. Learn to recognize left and right columns from the clues supplied by your system. Unless your system displays left and right columns on the screen, you must learn to recognize each to facilitate merging or pasting-up. One plan is to retain odd numbered pages for left-hand columns and even numbered pages for right-handed columns.

4. Place full-page-width materials only in the left-hand column. Full-width illustrations and tables must be placed in the left-hand column so that they can spill into the right. Also, remember to allow room in the right-hand column for the spill over.

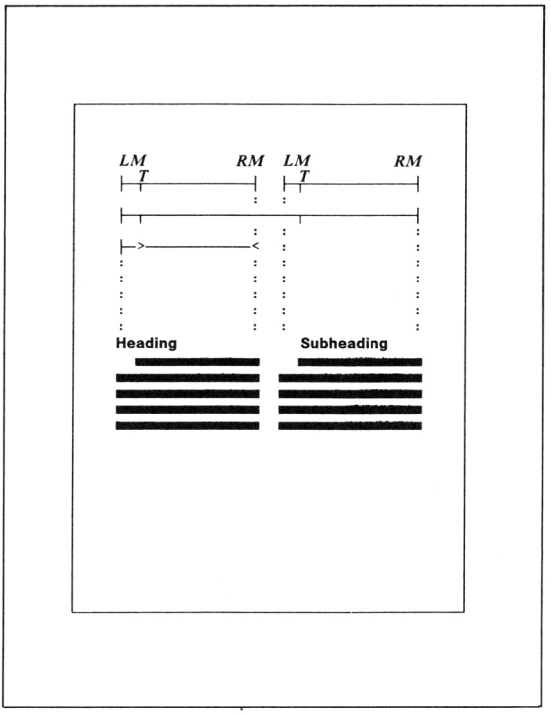


Figure 14.

Converting Type Specifications

Of all the procedures you can learn to give your work a professional look, converting fixed pitch to proportional spaced (PS) type may be one of the most important. PS type looks more like typesetting than fixed pitch type because it "closes up" text and allows the eye to interpret it more accurately.

But PS type is only available on some systems and some printers. Among character printers, only some take the typewheels whose characters are sized appropriately. To use PS type, use the following procedure.

1. Examine your use of formats: if you have been cheating by using spaces instead of tabs to achieve horizontal alignment, you must learn once and for all to use tabs. Why? Because in fixed-pitch work, the system allows the same spacing for a space that it does for any other character—one-tenth of an inch in 10-pitch, one-twelfth of an inch in 12-pitch, and so on. In PS work, the system allows the same space it allows for a lower-case "i," "t," or ",." PS allots more space for all other lower-case letters and all capitals, and destroys the spacing

achieved by the use of spaces in fixed-pitch. It does not, however, affect tabs or indents.

2. Examine your use of the screen. Very few screens can display PS work as it will print out because the raster design of the screen is actually in fixed pitch. Thus, your printouts will not be the same as the screen displays of the same work.

- *PS printouts:* PS printouts close up normal text (primarily in lower-case letters) by about 20 percent over fixed 12-pitch printouts. With more words to a line, paragraphs get shorter. Your format, which allows fifty-five characters in fixed 12-pitch, allows as many as sixty-six characters in PS printouts. Thus, to gain control of the column width in total characters, you want to change your format. At the same time, PS printouts tend to expand lines with capital letters by about 20 percent over fixed 12-pitch printouts. Your format, which allows fifty-five characters in fixed pitch, will fit as few as forty-five characters set in all capitals in PS work. By the same arithmetic, a line that is fifty-five characters in capital letters will require a sixty-six-character format to accommodate it in PS.

3. Examine your page lengths.

- *Vertical alignment:* With text expanded or contracted by 20 percent, your vertical alignment may also suffer. You may have created a dual-column document in which an illustration pasted into the left column extends into the right. Now the left column condenses during printout by one or more lines. Your right column, containing a heading in all caps (spreading into a second line during printout) and containing other significant numbers of capital letters, does not condense. The result may be that while the two columns are the same length on the screen, they are not the same length on the printout, and your paste-up will not work correctly.
- *Print routine:* You must also decide how to print text when some headings require one line and some require two. This is important for dual-column work, and for when unequal headings appear on facing pages. You have several options:
  - make the headings flush at the top of the heading, and let the text fall where it will after appropriate carriage returns;
  - make the headings flush at the bottom of the heading, and make the tops of all the paragraphs flush with each other;
  - install extra carriage returns after the one-line headings to ensure that both the tops of all the headings are flush, and the tops of all the paragraphs are flush.

(See Figure 15.)

Adding Headers and Footers

Running headers and footers lie outside the image area of the page. Page numbers always appear in a header or footer, as do headings and subheadings used as running headers. In addition, many people use a date, draft number, release, or

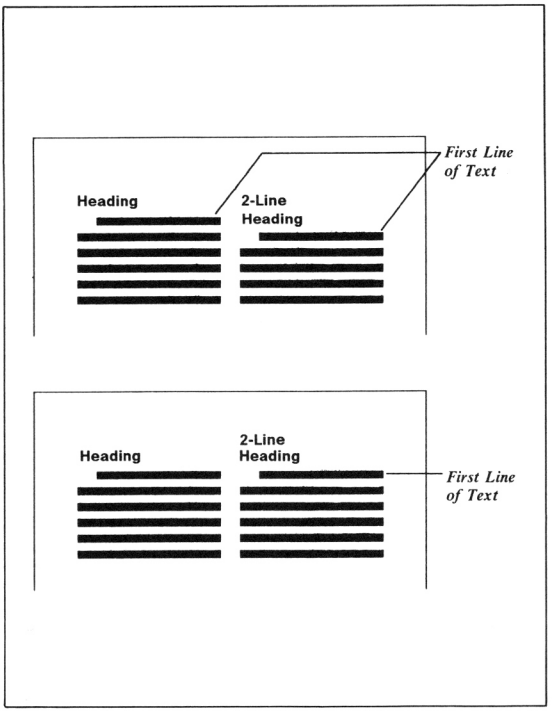


Figure 15.

revision number, or part number as part of a header or footer to keep track of the date and relevance of the information.

Thus far, our discussion has dealt only with the information inside the image area. It was based on the idea that the first line of text on a page might be the beginning of a paragraph or a heading or a subheading; and that the last line of text might be the end of a paragraph or a line of text or a caption of an illustration.

In discussing headers and footers, however, we venture into the top and bottom margins of the page. To install a header and footer on the pages of our report we have used the following procedure as an example.

1. Reexamine your print routine to learn how to control headers and footers installed during printout. Some word-processing systems provide this facility: when used, it provides consecutive page numbers applied to each page in the file, and a single, standard citation in either header or footer.

2. Expand into the top and bottom margin of the page. If you are unsatisfied with header and footer accommodations in the print routine, or if your software

does not provide it, you may have to install your own headers and footers. To do this, you must get control of:

- *Page length:* To install a custom header at the top of a page, you must increase the length of the page by forcing white space at the top into which the header can be inserted. To install a custom footer at the top of a page, you must force white space at the bottom of the page. Note that if your system places a page-length limit, this expansion may reduce the total amount of image area available to you. See Figure 14.
- *Print routine:* You must learn how close to the top and bottom of a sheet of paper your printer can print in both sheet-fed and form-fed printouts. Some printers cannot print the full length of the page; printer limitations force effective page limitations.
- *Software commands:* If your word-processing software allows you to develop software commands (macros), learn to write one that provides the header and footer information you need. Such macros can be used to develop customized page numbering, revision numbers, and other materials. In addition, a macro that can identify a heading or subheading, table title, or figure caption along with the page number can be used to develop front matter.

By now, you have seen that word processing can be used to capture keystrokes, edit the results, and turn text into readable pages.

Some word-processing systems and word-processing software packages make page design quite easy: they allow for the use of more than one format in a text file; they make those formats visible on the screen as the text; they support reformatting; they allow sufficient numbers of pages or lines in a text file to support technical documents; they support a method to link files in a disk index and in the print routine; they provide high line counts per page without penalty; they allow for easy insertion of white space and page breaks; they allow global replacement for formatting characters; and they display the results of your work on the screen so that you can evaluate the results.

Other systems and software packages provide some of these features, but not all of them, and may impose penalties on the user for some functions.

All word processing is limited by the software, the screen, the printer, and the word-processing developer's imagination. If your imagination for page design is broader and deeper than that of the person who designed your word-processing system, you have two choices: push your current system as far as it will go, or look for another system. Since ultimately every word-processing system has some limitations, and page design is the ultimate use of the system, you must make the most of the system you have. This means that you must experiment both on the screen and in the printouts. Sometimes a simple change can have big payoffs, and your imagination is the most powerful word-processing tool you have.

CHAPTER 8

Computer Graphics for the Technical Communicator

WILLIAM L. BENZON

A picture is worth a thousand words. But not if you can't find, buy, commission, or draw that picture. If you are not an artist, then you can't draw that picture. Or so you thought. As Gerswhin's Sportin' Life sang, "It Ain't Necessarily So."

Computer graphics has arrived. It has, in fact, been around for over two decades, but it has been available only to those who have access to large and expensive computers. With the introduction of the Xerox Star, the Apple Macintosh, the Commodore Amiga, and similar computers, things have changed. The Macintosh was the first relatively cheap personal computer to feature powerful graphics programs. And those programs, and others like them which followed for other computers, make it possible for anyone to be a graphic artist—not a first class graphic artist, but good enough to create simple charts and diagrams that are cleanly rendered.

These innovations and capabilities are important to technical writers because technical communication is visual as well as verbal. Technology often involves plans—for motors, for buildings, for microchips, for a set of chemical processes, as in refining oil—and such plans are often visual. The new computer graphics technology provides technical writers with easier access to tools for creating these diagrams and drawings. In some cases this graphics technology gives the writer the ability to create final images; in other cases it provides him or her with a tool which facilitates communicating with graphics artists by making it easy for writers to produce rough images.

In this chapter I will consider the capabilities of relatively inexpensive graphics packages, programs which can be run on personal computers. First comes a brief primer on technique—what you can do with desk-top graphics systems, then hardware is discussed, after that, software, concluding with comments on the implications of this technology for technical communication.

DO-IT-YOURSELF COMPUTER GRAPHICS: A PRIMER FOR THE TECHNICAL WRITER

Attitude is important. We tend to think that if we cannot render realistic images as well as Leonardo da Vinci, or Norman Rockwell, or at least the person who draws ads for the local department store, then we are not artistic. Fortunately, computer graphics allows you to sidestep this reaction by avoiding both sheets of paper and blanks. You don't work on paper, you work on a monitor screen. And you can start with images someone else has already prepared.

Assembling an Illustration

Figure 16, a highly schematic image of a local area telecommunications network, is a typical piece of technical illustration. It makes the point that local area telecommunications networks can handle voice (telephone), video (the monitor), and computer (the terminal, central processor, and printer) interaction. I created it for a consulting report on telecommunications. The images of the terminal, the telephone, and the man on the video monitor look as if they were

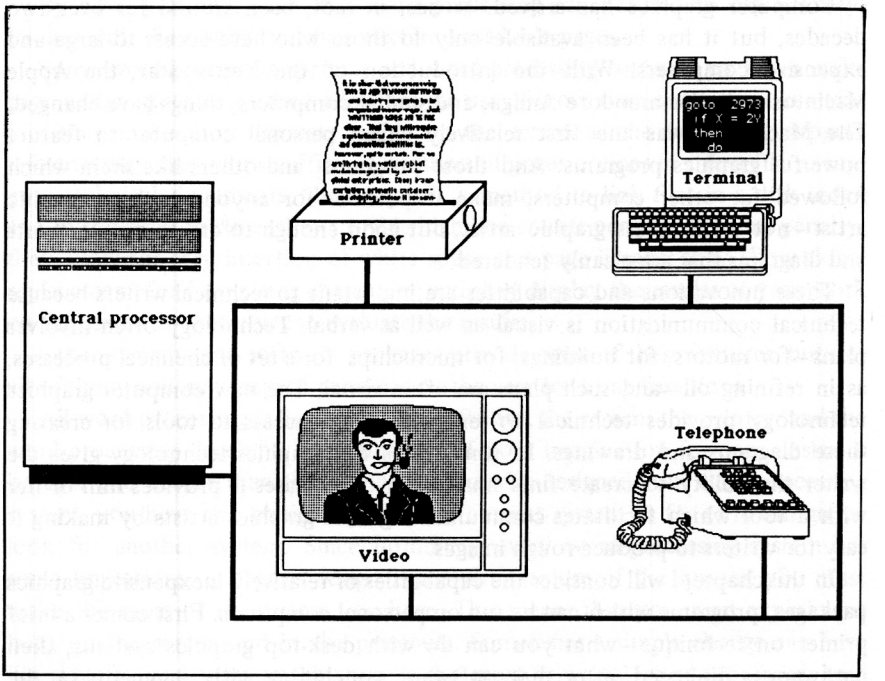


Figure 16. Local area network.

created by a professional illustrator. And they were; but not by me. I simply took those images from one of a large selection of images commercially available (these came from *McPic!*, Vol. 1, by Magnum Software). The terminal image came with a blank screen; I had to put the text on myself, which was quite simple.

I made the other elements of the illustration—the printer, the central processor, the video monitor, and the lines connecting everything together—with a graphics program. The central processor, for example, is created with rectangles, some of them filled with patterns, and a simple label. This is easily done, taking a few minutes to try variations in proportion and placement to see what looks best. If you are skilled in using manual tools—drawing board, T-square, triangle, pencil and/or pen—this is a simple diagram. However, if the proportions don't suit you or if the pen skips, then you may have to redraw the complete image. With a graphics system, there is no pen to slip and electronic redrawing is quick and easy..

Basic Visual Building Blocks

Drawing even a simple rectangle is much simpler with a graphics program than with traditional drafting implements. With manual tools you need four strokes of a pen or pencil, the T-square must be positioned three times (once for the top, once for the bottom, and once as the base on which you move the triangle to draw the vertical lines), and a triangle must be positioned twice (once for each side). Further, if you are drawing in ink, you then have to wait for the ink to dry and you have to be careful of the order in which you ink your lines. This is all rather simple stuff, but requires training in drafting.

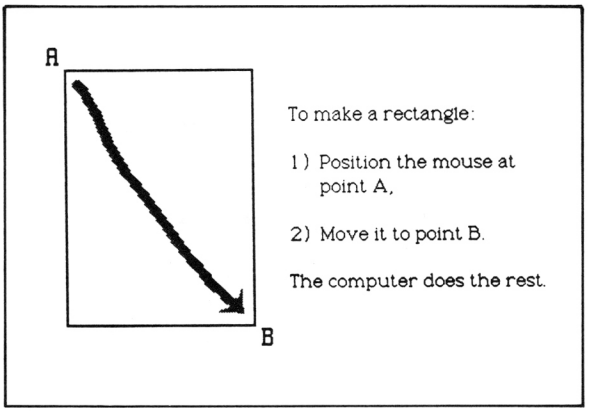


Figure 17. To make a rectangle.

But drawing the same rectangle with a graphics program is much easier, and no training is necessary—a six-year-old can do it. To draw a rectangle you make one motion with your input device (such as a pen or a mouse), in effect, from on corner of the rectangle diagonally across to another corner (Figure 17). That is, you use the input device to determine two diagonally opposite corners and the program inscribes the rectangle between them. Squares, ellipses, and circles are drawn in the same way. You can then create fairly complex images by appropriately combining these elementary visual building blocks.

Figure 18, which illustrates the computer programming concept of the linked list, consists of rectangles, straight lines (vertical, horizontal, and diagonal), arrowheads and numerals. First I created the basic building block, a long, narrow, horizontal rectangle, partitioned it toward either end and in the middle and then made copies of it and placed them appropriately on the page. And that's half of the job, one simple image, repeated over and over.

Figure 19 illustrates, in a highly stylized way, a certain pattern of neural circuitry (from the cerebral cortex). This image consists of two "assemblies," each of them called a column. The columns are identical. I drew one and simply copied it to make the other. Within the column we have repeated horizontal elements. I drew one and copied it to make the others. This simple procedure, draw one and copy it over and over, appears again and again in creating technical illustrations. Computer graphics tools are ideal for doing this sort of repetitive drawing.

Finally, consider the simple charts of Figure 20. The upper one is basically a bunch of horizontal lines and vertical rectangles. The lower one is a circle with several inscribed radii. These drawings are very simple to do. In fact, these particular types of drawing—graphs and charts—are so very simple that many graphics programs produce such images directly from input of the numerical data they depict (see the section on business graphics p. 132).

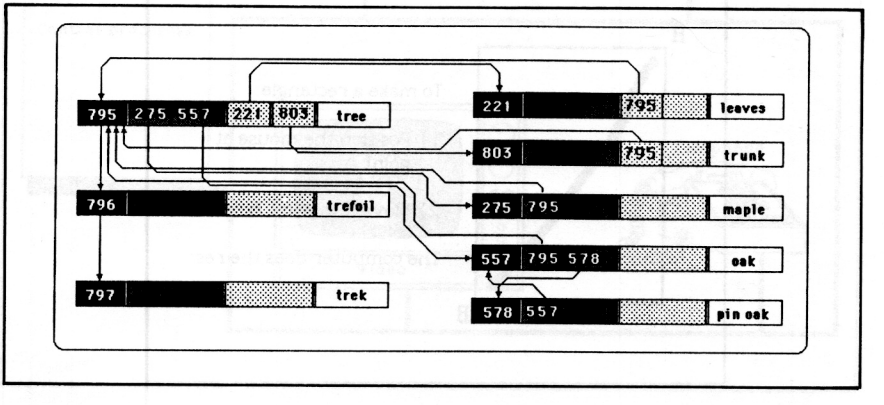


Figure 18. Linked list.

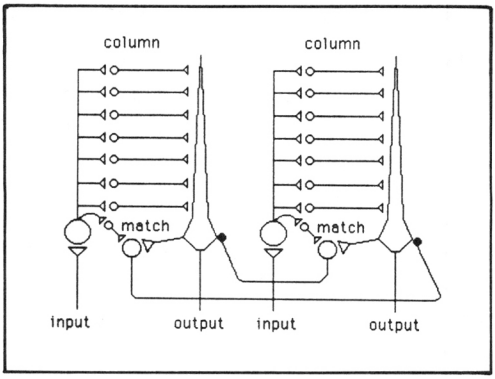


Figure 19. Neural circuitry.

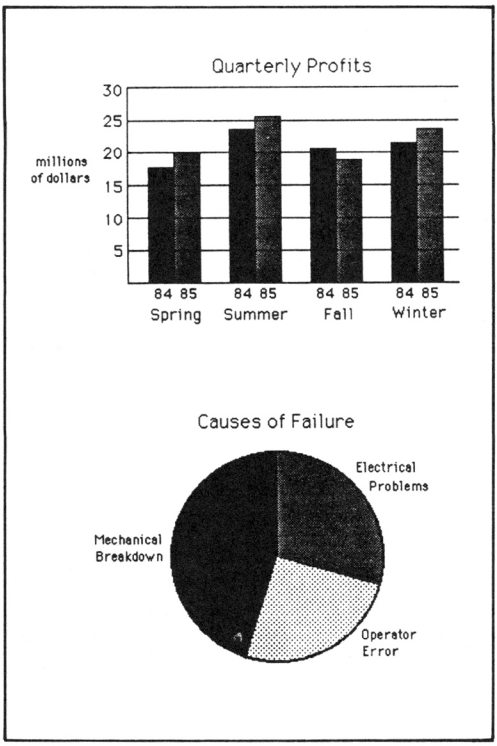


Figure 20. Graphs and charts are made of simple geometric shapes.

More Assembly: Building from Previous Illustrations

Now, consider Figure 21, which shows that a wide area telecommunications network links local area networks together. This is another example of "draw one and copy it." Only here I had to reduce the "seed" element in size. For, in this case, the seed element is another illustration. I simply took the diagram from Figure 16, the local area network, and reduced its size—very easy to do. I then made four copies of the diagram and put different faces (also from *McPic!*, Vol. 1) on the video screens. Then I drew a satellite dish (using only straight lines, rectangles, and one curve as basic elements), made three copies of the original, reversed the orientation of two of the copies, and placed them appropriately on the page. Finally, I drew the satellite and the "beams" connecting it to the dishes.

The central point is this: Once you have created an image, such as Figure 16, it is easy to modify that image and use it again and again, which I did in Figure 21. In this way you can build up a library of useful images and use these images in the same way you use maps, faces, machines, borders, and so forth, which you

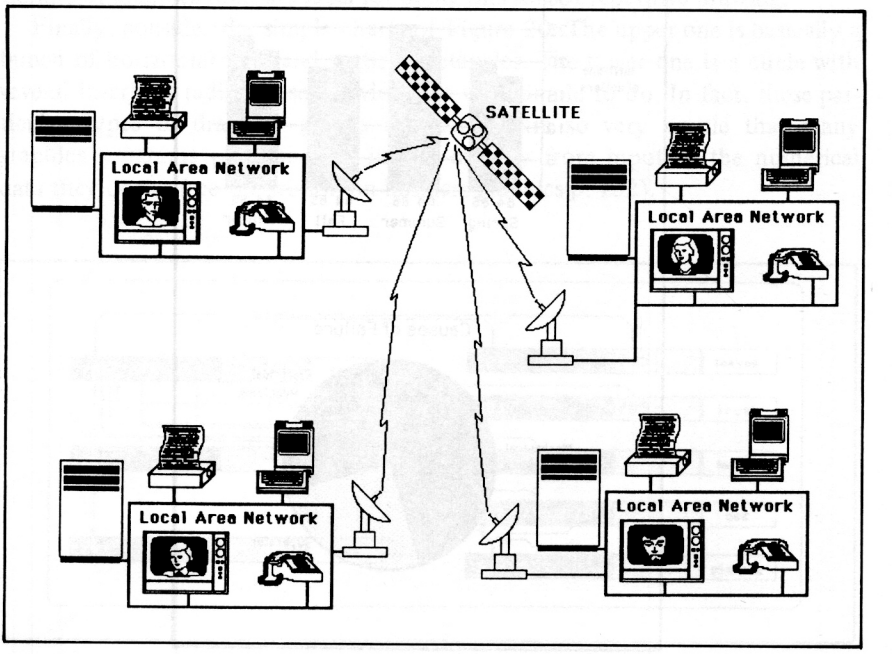


Figure 21. Wide area network.

have purchased from vendors or received from friends and colleagues. The use of such a system is thus cumulative, for images can be collected and then edited and reused.

Images of Abstract Concepts

Figures 22 and 23 show how to use images to illustrate abstract concepts. Figure 22 (with faces from *McPic!*, Vol. 1) is in the same style as Figures 16 and 21, but it's subject matter is abstract. Figures 16 and 21 depicted, in a highly stylized way, telecommunications hardware, which are physical objects. Figure 22 isn't about physical things. It is about information, which is not tangible. The diagram shows that, in an organizational hierarchy, information becomes abstracted as one moves up the hierarchy. At the bottom we have volumes of highly detailed information about a construction project (the labeled boxes above the highly stylized cityscape). At the top we have a three-page report for the chief executive. Again notice the many repetitive elements in the illustration. In fact, there may be too much repetition—all those engineers having

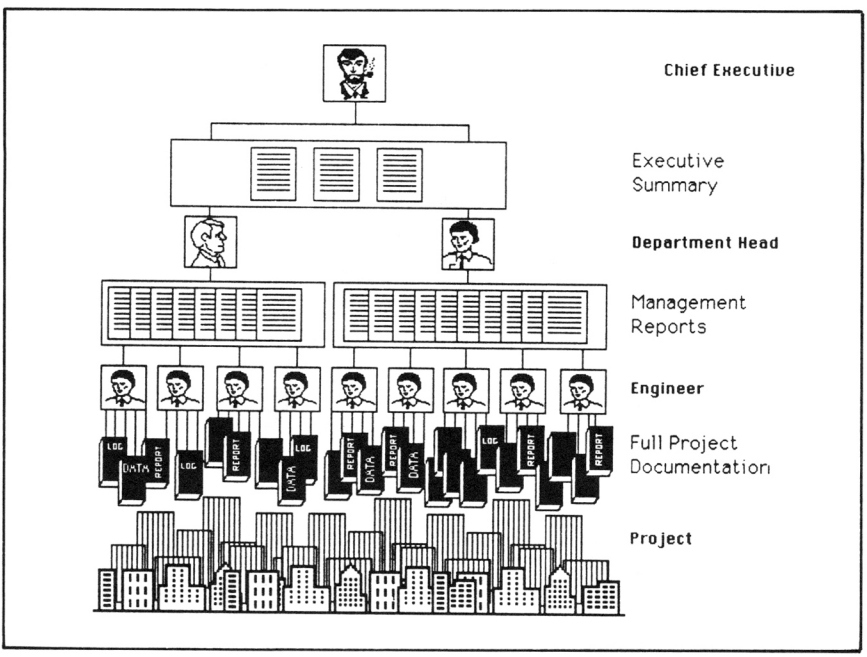


Figure 22. Abstraction in the organizational hierarchy.

the same face might convey the wrong message. If this is a rough drawing, to be finally rendered by a graphic artist, the artist will reduce the redundancy by using different faces.

Figure 23 is from the same report which Figures 16 and 21 are from, but its style is fundamentally different. It is an exceedingly simple image and it makes a simple point. In the modern shipping industry, ships, goods, and the containers must be tracked independently because the relationship of particular goods, containers, and ships, is only temporary. Even such simple images provide strong communication aids for they make ideas more concrete; they give the visual mind something to work with even as the verbal mind digests.

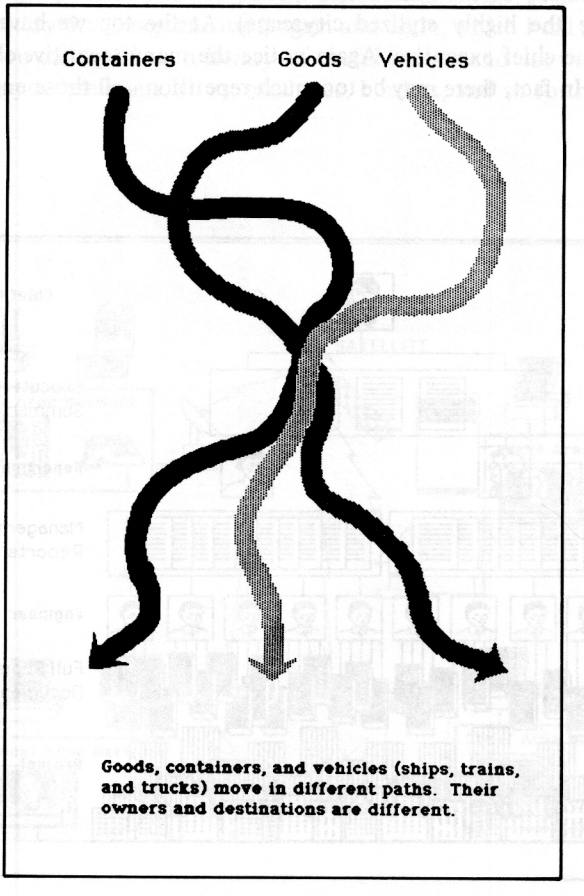


Figure 23. Goods, containers, vehicles.

Logos and Graphic Designs

Figures 24 and 25 are less typical of the applications a technical communicator faces, but, aside from the fact that technical communication is so diverse that virtually any type of image will have some use, both illustrate how easy a basic design can be varied. Figure 24 is a set of logos based on the initials "FLN." We have three basic versions of the letters, each of which had to be specially drawn. The version at the top is obviously the most standard while the other two involve considerable distortion of the basic letter shapes. Once I had created a visually interesting or pleasing set of letters I rendered each set in various ways—the variations presented in Figure 24 represent less than a quarter of the ones I tried. Creating the theme, the basic set of letters, was tricky. Since this problem is aesthetic—finding interesting shapes—it would be equally difficult no matter what medium I used. But, with the computer, it was easy to create the variations on the theme.

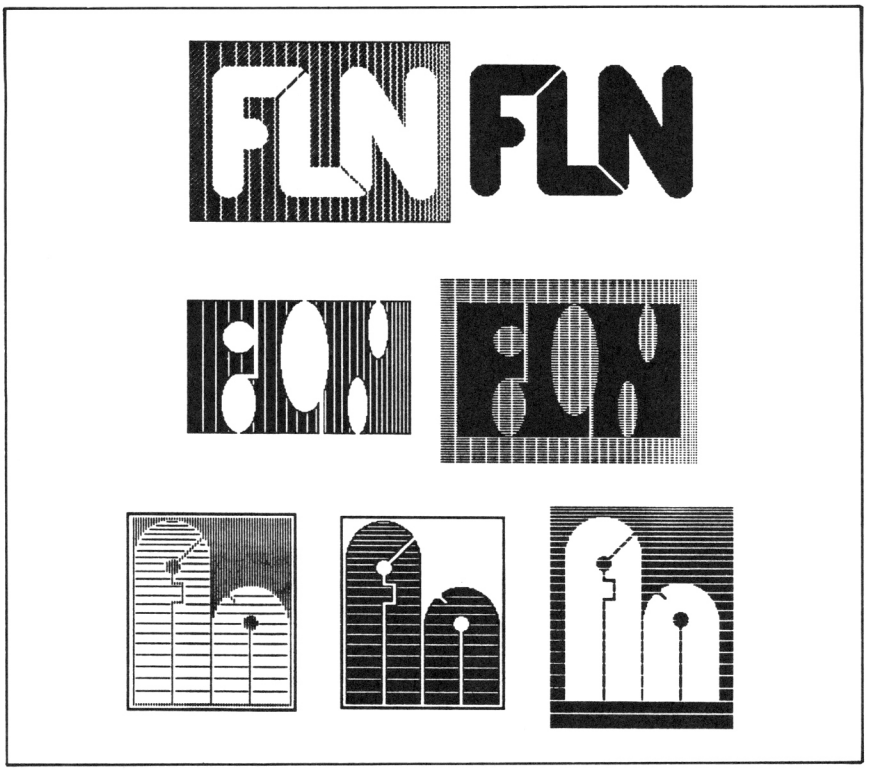


Figure 24. Logo designs based on the letters "FLN."

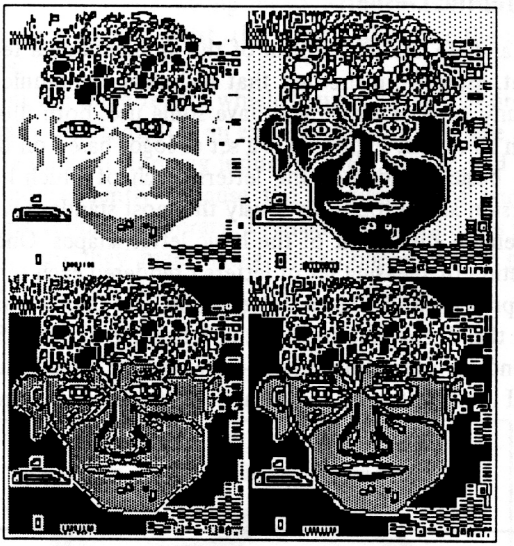


Figure 25. Four easy faces.

Figure 25 started as a small crude face, which was simply part of a larger pattern. I simply removed the face from the larger pattern (and destroyed what was left) and enlarged it. I then created several copies of the enlarged face and varied the copies by "painting" various areas with different patterns (in this case, simply different densities of dots) which are defined by the graphics program.

Digitized Images

Finally, some graphics programs can work with pictures input through an image digitizer. You can modify these pictures—shrinking them, enlarging them, pasting them into pictures you've drawn with the paint tools. Figure 26 is a digitized photograph while Figure 27 is a digitized line drawing. Neither is particularly wonderful—the digitized photograph is coarse and the digitized line drawing has many ragged lines. But they are quite adequate for rough graphics and they can be improved a bit through electronic editing.

The Use of Computer Graphics

These images may be adequate for in-house publications and presentations in situations where company image isn't so crucial. Perhaps, however, the most plausible way for a technical writer to use graphics technology is to develop rough art as the basis for interaction between the writer and an illustrator. You

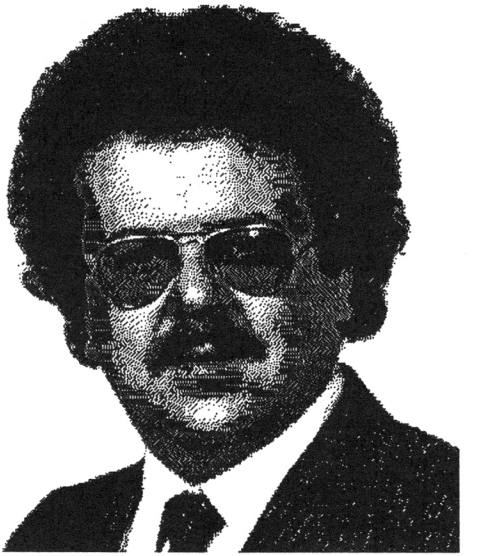


Figure 26. Digitized photograph.

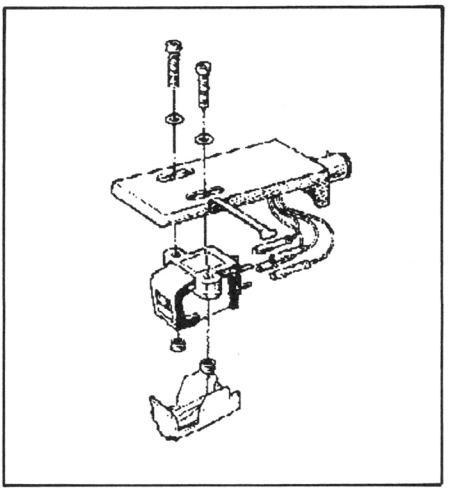


Figure 27. Digitized line drawing.

can use a graphics package to create a rough idea for an illustration. The idea can involve any of the techniques I've discussed, from simple geometric forms to complex digitized images and combinations of various techniques. It could be an illustration of a piece of equipment or an abstract concept. It could be intended for a repair manual, an annual report, an applications story intended for public news media. Or you could use graphics technology to develop a story board for an audio-visual presentation or a film.

The illustrator can take your rough concepts, make suggestions, and perhaps rough out concepts suggested by your work—all electronically. When you finally arrive at a concept you both like—it serves its communicative function and can be executed within the budget for this particular publication—the illustrator can render it in traditional media—a line drawing, a color painting, a photograph. In this way writer and illustrator can interact through a common language of images. The technical writer, having mastered the art of translating technology into prose, can now move on to translating both technology and prose into images.

HARDWARE AND SOFTWARE

Any computing system consists of hardware, physical devices, and software, the instructions and data which are manipulated in those physical devices. In order to understand the basics of computer graphics it is necessary to understand the fundamental principle of computer graphics:

- **Pictures are made of dots.**

Happily, this principle is fairly common knowledge. It is, for example, also the principle underlying the halftone process used in printing both color and black and white pictures in newspapers, magazines, and books. Each dot is a picture element, or *pixel*. All of the computer graphics systems I'm concerned with handle images as planes of pixels. The *resolution* of the system is defined in terms of the number of pixels which the system can handle in a given area. The more pixels (dots), the higher the resolution.

The capacity of a graphics system is a function of how many pixels it can manipulate and how much information it can manipulate about each one. Let's consider a simplified example. Assume we have a system which can manipulate up to 131,072 *bits* of information for each image. If we are satisfied with a monochrome image, then each pixel needs only one of these bits to encode its state; the pixel is on or off (to think in terms of displaying it on the screen), black or white (to think in terms of the color of a printed image). Thus we can encode 131,072 pixels with our 131,072 bits. This would correspond to a square approximately 362 pixels on a side.

However, if we wanted each pixel to be one of sixteen different colors, or one of sixteen shades of gray (which could range from blackest to whitest),

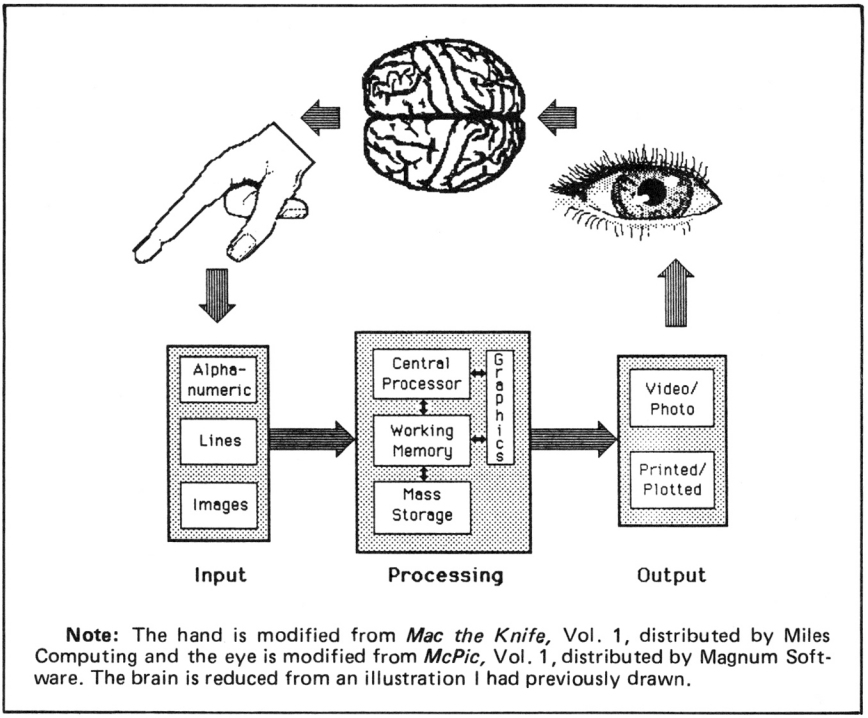
then we need 4 bits to describe the state of each pixel. Hence the largest image we can handle would contain 32,768 pixels, corresponding to a square approximately 181 pixels on a side. The colored image the system can handle is thus smaller than the black and white image it can handle. If we wanted to be able to use more colors, then the maximum image size shrinks still further. Alternatively, of course, we could use a bigger and more sophisticated system in order to handle relatively large color images.

The basic point is this: In any graphics system there is a tradeoff between resolution and color. The more resolution you want, the fewer the number of colors you can handle. Conversely, the more colors you want, the lower the resolution of your image.

What Is a Graphics System?

Computer graphics systems (see Figure 28), like all computer systems, consist of:

- an input device, or devices, which get pictures, diagrams, and so forth into the system;



Note: The hand is modified from *Mac the Knife,* Vol. 1, distributed by Miles Computing and the eye is modified from *McPic,* Vol. 1, distributed by Magnum Software. The brain is reduced from an illustration I had previously drawn.

Figure 28. Schematic of graphics system.

- something to process the images, that is, the computer proper; and
- an output device, or devices.

First I want to say a little about the computer proper and then move to a more extensive treatment of input and output devices. After that I'll move on to software.

The Graphics Processor

Graphics processing is more demanding than text processing. The entire text of this chapter occupies a file of about 44,000 bytes (1 *byte* = 8 bits); and, when printed out on double-spaced pages of standard size (8½ by 11 inches) paper, the text is about thirty-one pages long. The image files vary between about 6,000 and 21,000 bytes, with most of them being toward the upper end of the range; the total size of the image files is about 209,000 bytes and the average size is about 16,000 bytes. Most of the images each require a page. Thus the computer has to handle about 16,000 bytes to store one page of graphics while it needs only about 1400 bytes to store a page of text. The disparity would be much greater if the images were in color.

Thus systems for graphics must be larger than systems for text. You need at least 128 kilobytes of working memory (RAM) for a graphics system, preferably more. Some computers, such as the IBM PC, require special graphics boards to handle screen graphics (hence the "graphics" box in Figure 28), whereas others, such as the Macintosh, do not. In addition, if you do a lot of graphics you might want a hard disk, instead of floppies, for mass storage. Hard disks can store more files and they can load and retrieve them from working memory more rapidly, making them more convenient to work with.

Input Devices

For any application, you want an input device which lets you deal with information in a natural way. Keyboards are ideal for entering alphanumeric characters, which are the basic visual elements making up a text. The vast majority of graphic images, however, are not alphanumeric characters. They are made up of lines—long, short, straight, jagged, curved, narrow, fat, and so forth. And so you want an input device which can deal in lines. The following list is not exhaustive, but it does indicate the most useful devices:

- mouse
- light or touch pen
- digitizing pad

The mouse, a small device about the size of a cigarette pack, is attached to the computer by a flexible cable. You place the mouse on a smooth flat surface and move it around. The mouse transmits signals to the computer about

its motion and the computer uses that information to draw lines on the screen. If you move the mouse in a straight line, a straight line will appear on the screen. If you move the mouse in a graceful curve, a graceful curve will appear on the screen. If you move the mouse in a rapid and random pattern, spaghetti will appear on the screen. The mouse generally has one, two, or three buttons on it which you can use to send signals to the computer. The most important signals are: Start Drawing and Stop Drawing.

Light or touch pens are used like a pencil. Instead of drawing on a piece of paper you "draw" directly on the monitor screen. Light pens work with ordinary monitors, but you need a special touch sensitive screen to use a touch pen. And, if you do have a touch screen, you can also use your finger tip for drawing on it.

Digitizing pads are electronic drawing boards. You draw on a special board, which has sensors to detect stylus pressure, just as you would draw on a piece of paper mounted on a drawing board with a pen or a pencil. The board transmits your drawing (that is, the motions of the stylus) to the computer, which in turn transmits it to the screen—so rapidly that the transfer appears to be instantaneous. Digitizing pads are the most versatile of these three input devices. The resolution of the mouse and light or touch pens is relatively coarse—they do not respond precisely to small changes in position, therefore you do not have very sensitive control over them. This makes it difficult accurately to render detailed images. Digitizing pads can be coarse or fine—but here, as everywhere, you have to pay more for precision.

Using digitizing pads is also much more natural than using either mice or light or touch pens. Mice simply don't allow you to use your fingers or, for the most part, your wrist in controlling the line. You control the line with your elbow and shoulder, which is fine for straight lines and for practicing ovals in grade-school penmanship classes; but otherwise . . . let us just say that it takes practice. The pens do give you finger and wrist control, but the vertical drawing surface is uncomfortable. With digitizing pads you have all levels of muscular control over the line—shoulder, elbow, wrist, and finger—and a comfortable drawing surface.

Further, digitizing pads can be used to trace images. You simply tape a drawing or diagram to the pad and then trace over it with the stylus. The outline will be transferred to the monitor screen. You can't really do this with either the mouse or the pens.

Thus, the digitizing pad is the most flexible device for entering graphic information, but it is mandatory only for the most demanding applications—rapid and easy entry of detailed images, especially images with many curved and slanted lines. If you have time and patience, even detailed images can be entered with a mouse or with a light or touch pen. You may not need a digitizing pad if your images aren't exacting; for example, rough drawings to be refined by a graphics artist, simple charts and graphs. In any case, any of these input devices can (and will) also be used to select commands from menus displayed on the screen.

Another type of input device, the image digitizer, is entirely different from the other types. These digitizers enter complete images automatically and use either video or optical technology. The video digitizer takes the output from a video camera or a video recorder (though not all digitizers handle both), and recodes it into a set of pixels, which can then be displayed on the monitor screen. These images are comparable to the image in Figures 26 and 27, which were, however, produced with an optical digitizer (p. 125). With a video digitizer you can input drawings, diagrams, charts, graphs, or photographs which already exist in hardcopy form. You can also input anything which you can place in front of the camera. However, the digitizing process is not instantaneous, so you cannot use a video digitizer to take a snapshot of something in motion (if you want to do that you would have to make a video tape of the moving object and then input individual frames from the tape). There is no drawing of any sort involved here. This is, in effect, electronic photography.

With the most readily available optical digitizer, you replace the ribbon cartridge for your printer with a cartridge containing an optical scanner. You place your image into the printer, like you would a single sheet of paper, and then scan it, one line of pixels at a time. This takes considerably longer than a video digitizer (5 to 20 minutes per image), but you don't have to worry about lighting, as you do when getting input from a video camera. The more traditional (as if the word "traditional" can be meaningfully applied to such new technology) optical digitizer is a flat-bed digitizer. You place the image on a flat bed and it is then scanned by the optical sensor.

These digitizers won't actually help you create images, but they allow you more effectively to use graphic materials already available. And once you've digitized an image you can manipulate it with your software just as you can any other image. For example, suppose you wanted different faces for the engineers in Figure 22. You could take a photograph of a small group (from a magazine, newspaper, or your own snapshot) and digitize it. Then you could electronically cut each face from the digitized photograph and paste them into the appropriate boxes.

Output Devices

Graphics output takes two forms, video and hardcopy. Let's begin with video output, which is, of course, the monitor screen which has become standard for all computer terminals (it wasn't always so—not when card readers and teletypes were the primary input devices). You send characters in through the keyboard and lines and shapes in through, shall we say, a mouse. And both appear, instantaneously, on the screen. That immediate feedback from the screen is so important to our input activity that we tend to think of the monitor as part of the input system. And so it is. But, technically, it is an output device. You don't put information into the computer through the monitor; you use the monitor to find out what has been put in by other means and what the computer has done to it.

Video output involves both the monitor which displays the image, and the graphics hardware which drives the monitor. That graphics hardware may be as simple as a memory board, or it might be a complete computer system specialized for graphics.

Monitors can be either low resolution or high resolution, color or monochrome. A low resolution monochrome monitor is cheap. A high resolution color monitor is expensive. Further, high performance monitors require higher performance graphics hardware in the computers that drive them. Since, as we've seen, every pixel on the screen corresponds to a memory element in the computer, the more pixels on the screen (that is, the higher the resolution), the more memory elements the computer has to have to handle the image. And, the more colors you want to be able to choose from, the larger the memory element needed for each pixel. Thus there is a close relationship between the quality of your monitor and the power of the computer needed to support it. Performance has its price.

And sometimes it is worth the cost. Especially if you want to work with color. While color printers are available, their color doesn't rival that of even a cheap color monitor. If you are using a graphics system to produce images which will be used for color slides or color plates, the simplest way to get a hardcopy image is to photograph the monitor screen. (It is possible to print an image directly on film, but the technology is expensive and necessary only in the most demanding applications.) The brighter your monitor screen and the higher its resolution, the better your photographs.

Another output device is the printer. There are more different kinds of printers than kinds of animals in Noah's ark, but you don't, fortunately, have to take them in pairs. The ubiquitous letter-quality daisy wheel printer, the staple of word-processing applications, is not, however, very useful for graphics.

The dot matrix impact printer is much more useful for graphics. As its name implies, the dot matrix printer employs our basic graphics principle, that any image can be represented as a pattern of dots. Instead of having a print head with a complete set of alphanumeric characters arranged in a daisy wheel, for example, the basic dot matrix printer has a print head consisting of individual pins which are directed against the ribbon in appropriate patterns. Letters can be formed, and so can arbitrary shapes—the lines, dots, and patterns needed for graphics printing. As is the case with monitors, the higher the resolution, the more expensive the printer. Resolution of 60 to 70 dots per inch is commonplace and cheap; you need at least 120 to 150 dots per inch for reasonable graphics. Note, however, that you can photographically reduce an image for reproduction. Thus an image which was printed out at 70 dots per inch can be photographically reduced 50 percent, giving you a resolution of 140 dots per inch for reproduction.

The dot matrix principle is also used in ink jet and laser printers. Ink-jet printers squirt a small stream of ink (actually ink droplets) directly onto the

paper, eliminating the mechanical pounding of impact printers. They are thus quieter than mechanical printers.

Laser printers are the most versatile and the most expensive—about 10 to 100 times as expensive as impact or ink-jet matrix printers. They work on the same principle as electrostatic copiers and have a resolution of 300 or more dots per inch. Professional phototypesetting equipment, by contrast, has a resolution of 1500 to 5000 dots per inch. Laser printers are fast, with the slowest printing high quality characters at about 300 per second (where the best mechanical dot matrix printers can print their highest quality characters at about 90 characters per second).

All of these printers are basically monochrome printers—black and white. You can get impact dot matrix printers which take four-color ribbons so that by suitably overprinting different colored dots you can get sixteen or more colors, but the quality is not high—not comparable to what you can get by photographing a decent monitor. Similarly, you can get ink-jet printers, which are better than impact printers, but still not as good as photographing a monitor.

Finally, we have to consider pen plotters. These move small pens over paper and, unlike any of the printer types, can accommodate fairly large paper sizes—twenty-four or more inches across. These are suitable only for line drawings and, unlike the printers, are not at all suited for printing large quantities of text. They do, however, print smoother lines and so are better for drafting applications. They also print in color.

Types of Software Packages

We can divide graphics packages into four basic types: business graphics, paint systems, drafting systems, and page make-up systems. Business graphics systems, for the most part, allow you to make a variety of charts and graphs in standard formats. Paint systems and drafting systems allow you to create your own images, but there are significant differences in the way they store their images, which in turn affect the capabilities of the systems. Roughly, if you want to do free-form shapes—such as in cartoons or in logo designs—then a paint system is your best bet. If you want to do images such as flow charts, building plans, or gear trains, then a drafting system might be better. Finally, page make-up systems allow you to mix graphics and text in a wide variety of formats so that you can put documents together and print them out on your mechanical dot-matrix, ink-jet, or laser printer.

Business Graphics

The basic business graphics package allows you to produce charts and graphs and is designed to work without the user having to do any "drawing" at all. You enter the numbers you want to graph, either from the keyboard or from a file used by the application program which produced the numbers (such as a

financial spreadsheet program) and then choose how you want to display the numbers—a line graph, bar chart, pie chart, or appropriate variations of these types. Figure 20 shows the sort of output you can get. If you have a color screen, and a color printer, you can have the graph done in color. You choose the pattern or color for each area, choose the labeling, and the computer produces your chart. If you don't like the first version you can change it, easily. If you started with a bar chart and decide a pie chart would be better, you can produce one very easily. And you never have to look at messy drawings. The screen image is always clean and so are the printouts.

These packages don't require any graphics skill at all. The program makes your graph or chart automatically. Once you have something you like, you can photograph the screen to produce slides or you can have the graph printed out and use it as part of a hard copy document, or you can use it for making a viewgraph foil or a printing plate. Even where the output of the printer isn't good enough for the final chart, such packages are useful because they allow you to get a fairly good idea of what kind of chart you want without first having to go to an artist. Once you know what kind of chart best depicts your data, you can take the print-out to an artist for a high quality rendering.

Paint and Drafting Systems

Paint and drafting systems are the most generally useful packages for technical communicators. These systems most commonly require a mouse, or a digitizing pad, for input and provide you with a variety of "tools" for creating and editing images. Paint systems store images as patterns of dots while drafting systems, which are basically CAD (computer aided design) applications, store images as a collection of small functions for drawing geometric shapes. Any image which can readily be described as a combination of simple geometric shapes—squares, rectangles, hexagons, circles, ellipses, etc.—can be rendered in a drafting program. More irregular images are better suited to paint programs.

The patterns in the logos (Figure 24), for example, aren't very naturally treated as collections of simple geometric shapes, and the shapes in the faces (whether in Figures 16 and 21, 22 or in 25) are almost impossible to describe in that way. Hence those types of images cannot be created in a drafting program (though it is sometimes possible to create such images in a paint system and then move them into a drafting system). Note however, that if we reconceptualize Figures 16 and 21 so that we don't need the faces or the telephone and terminal images, which would also be difficult to handle in a drafting program, then we could render them through a drafting program. It would be quite easy, for example, simply to replace the telephone and terminal images with appropriately labeled boxes.

The images in the other figures—the nervous system (Figure 19), or the linked list (Figure 18), or even the charts from Figure 20—can be rendered in

drafting systems. In general, anything which looks like a blueprint can be rendered in a drafting system while anything which has many irregular shapes and lines is more appropriate for a paint system.

And, for their range of shapes, drafting systems are more powerful than paint systems. One capability which is particularly important is the ability to make a very large drawing. The paint systems in our performance range are limited to images which can fit on a single sheet of standard letter-sized paper (if the images are to be photographed from the screen then, of course, they can't be any larger than the screen). Drafting systems can handle much larger images by storing them on a "virtual" page, a page which is abstractly defined within the computer. You can only look at a fragment of such a page through the monitor, but it is easy to "move" the page about so you can view various sections of it. These large drawings can also be printed out, though it may be necessary to print them out section by section and then assemble the sections by taping them together.

If, for example, you must routinely make diagrams of large organization charts, or very large diagrams in the style of Figures 21 (and can live without a realistically rendered telephone) and 18, then a drafting program is more suited to your needs than a paint program. If exact dimensions and proportions are important to you, regardless of the overall size of the diagram, then you need a drafting program, which is equipped with electronic rulers (paint programs aren't so equipped). In general, however, drafting programs are intended for designers and draftsmen, engineers and architects, and many of them are intended for specific design applications—electronic circuit board design is a very common application area.

There is one final feature of drafting programs which is worth mentioning. Some of them can handle three-dimensional images. Now, we have to be careful here. Consider, once again, the telephone and the terminal (or even the printer) from Figure 16. These are depicted in three dimensions, but those three dimensions have been projected onto a two-dimensional surface by the person who drew the image. The paint system doesn't have any information about the third dimension—it just has a two-dimensional array of dots. Drafting systems, however, because they store images as a collection of small shape-defining functions, can store three-dimensional information; all that is needed is that the shape-defining functions handle three coordinates (X, Y, and Z) rather than only two (X and Y). With a three-dimensional graphics system you can examine any of the conventional two-dimensional views of an object—top, bottom, front, back, left side, or the right side, or you can view the object in a two-dimensional projection of all three dimensions. The most sophisticated systems will even allow you to rotate such a two-dimensional projection so that you can examine the object's surface as though you were moving around the object. Like two-dimensional drafting systems, three-dimensional systems are intended primarily for designers.

What all of these paint and drafting systems have in common is the capability freely to edit and modify images. None of them will turn you into a graphic artist.

None of them will give you the manual skill needed to draw a realistic face, or a complex machine part in isometric projection. Nor will they give you the imagination you need to design an attractive logo. But they will all make it much easier to explore visual ideas. Because it is so very easy to vary images, you can very quickly create a half-dozen versions of a diagram and then select the most attractive one. If you had to use manual tools, you wouldn't have time to do more than one version, and you would be committed to that one version no matter how good it looked. A skilled graphics artist can produce a fine drawing the first time through, but that is, in part, because he or she can explore various possibilities in his or her imagination, without ever having to make a drawing—that takes training and experience. With a suitable paint or drafting program you don't need to be able to imagine alternatives, you can produce them one after the other and then examine them. Nor do you need to be afraid of the mistakes you will inevitably make if you explore visual ideas. For, while a mistake on a hard copy image may cost you hours of work, a mistake in an electronic image can be fixed in minutes. By eliminating the difficulties associated with the manual production of images and diagrams, paint and drafting programs make it much easier to explore visual ideas. That ease is dramatic enough that visual thinking can readily become part of a technical writer's repertoire.

Page Make-Up

Page make-up programs allow you to combine images and text into a completed document which can then be printed out on an impact dot-matrix, ink-jet, or laser printer. In applications where the print quality is good enough (an in-house newsletter, for example) these programs can be used for complete typesetting, layout, and design. Coupled with a laser printer, they could even be used for print runs of up to about 500 copies; beyond that it would be cheaper and more efficient to have them printed on a press. These programs could also conceivably be used for rough layout and design in applications where the printed output, even of a laser printer, isn't good enough for producing camera-ready mechanicals.

The real reason for discussing these programs, however, is for what they portend for the future. Their limitations stem from size—the monitor screens aren't large enough to permit viewing of a complete page, nor is their resolution fine enough to handle high quality images, and the printers aren't precise enough to produce very high quality alphanumeric characters or good photographs. These limitations will surely disappear as hardware becomes cheaper and more sophisticated. When that happens it will be possible completely to design high quality documents electronically—text, typesetting, images, and layout. The documents could then be distributed electronically or through traditional hardcopy media.

IMPLICATIONS

Now, imagine a publications group in which everyone, writers, artists, designers, editors, works on such a facility, with all of them linked together in a local area network (for example, as diagrammed in Figure 16). What could that do to the publication process? Writers could create simple graphic images which could then be refined by artists. Writers could also examine preliminary layouts to insure that images are positioned next to the appropriate text. And professional artists could produce images which writers and editors could use for their own illustrative purposes.

With everyone having quick and easy access to one another's work it is not hard to imagine that traditional job roles will have to be redefined. While artists may well be relieved at no longer having to endure the tedium of rendering simple charts, they might also feel that the writer is beginning to encroach on their territory. For, if writers can now invent rough concepts for illustrations, then the illustrator may lose some creative freedom and become merely a functionary rendering someone else's graphic concepts. Or, more optimistically, the illustrator can use the writer's suggestions as just that, suggestions, and treat them as a "head start" toward a mature visual concept. If this happens then we might see a dramatic increase in the visual appeal of technical documents and a concomitant increase in their communicative effectiveness.

APPENDIX A: BITS

The bit is the standard unit of information. It corresponds to the answer to a two choice question, one where the answer is one of two alternatives, such as "yes" or "no," or "black" or "white." If you have monochrome display, with no gray levels, then each pixel can be encoded by one bit of information. Is it black or white? Now, if we have sixteen colors to choose from (or sixteen levels of only one color), then we need 4 bits, that is, four yes-no questions, to encode the state of each pixel. Number the colors from 0 to 15 and then ask, four times, Is the color in the upper or the lower half? For example:

1. Is it in the upper or lower half (of 0-15)? **Lower.** That is, 0-7.
2. Is it in the upper or lower half (of 0-7)? **Upper.** That is, 4-7.
3. Is it in the upper or lower half (of 4-7)? **Upper.** That is, 6-7.
4. Is it in the upper or lower half (of 6-7)? **Lower.** That is, 6.

Thus, four two-choice questions have allowed us to determine which of sixteen alternatives we are dealing with. The color, or gray level, of this pixel is the one encoded by the sequence: Lower, Upper, Upper, Lower. If we symbolize Lower with a zero (0) and Upper with one (1), then that color becomes 0110 in binary notation, which is the basic coding scheme of all digital computers. If we had to choose from among thirty-two alternatives, then we'd need five bits, six bits for sixty-four alternatives, seven bits for 128 alternatives, and so on.

APPENDIX B: HOW TO FIND OUT MORE

Information on graphics technology is readily available. The *Whole Earth Software Catalog*, which was first published in 1984 and is being updated regularly, is a good place to get an overview of graphics software packages for personal computers; it also has some discussion of hardware. All of the major personal computers have at least one magazine devoted to them; these magazines review hardware and software and feature applications articles. *Byte* is the major monthly magazine on personal computers and it has an issue on graphics every other year or so. *Infoworld* is the major weekly magazine; it is noted for its excellent reviews of hardware and software. If you want to see what the really powerful graphics technology is about – the technology which requires superminis, mainframes and supercomputers – check out *Computer Graphics*, the journal of SIGGRAPH (Special Interest Group on Graphics, of the Association for Computing Machinery); their third issue (of 4) of the year is the proceedings of the annual SIGGRAPH Conference and it is a stunner.

CHAPTER 9

Text Preparation and Transmission for Word Processing

DAVID T. ORR

With computer technology cheap enough to be available in most homes, few places of business have not made investments in computers of some kind. Furthermore, word-processing software can be obtained for virtually all computers, whether mainframe, minicomputer, or microcomputer. For all publication needs, but for technical publications in particular, the selection of equipment and software represents a long-term investment that informs both the product and the working relationships that underlie the production of texts. In this chapter, four major issues are examined. First, the methods for inputting characters are considered. Second, the role of editing and the shape of editing groups in preparing texts for publication is examined. Third, the variety of ways in which texts are coded for typesetting is explored. Finally, some of the methods for moving texts electronically are summarized. All of these are major thresholds in the production process of all texts.

CAPTURING CHARACTERS FOR WORD PROCESSING

Writers at Keyboards

Original keystrokes can be "captured" from the author through electronic storage or transmissions. That's called "author original keystrokes," or "A-OK." It is undeniably the best way to get text into an electronic form without editors having to type it in themselves. It is doubly useful when your author (or authors) is located at a remote site. Often text is transmitted and typeset on the same day.

Today, most writers work on some kind of word processor. Not only are these authors willing to take the time to transmit their texts, but they are usually

willing to include editorial codes. After all, the author would prefer to indicate a first degree heading (a topic headline) in the same way the editor does.

Publishers are now aware of the growing number of electronically prepared manuscripts. However, most publishers have not discovered how to set up a computer station and telephone number (toll free), how to leave it running all night, and what to do with the material they receive. They are not sure if it will save time and money.

It is not yet a good idea to transmit final galleys in "soft copy"—particularly if there are math equations and Greek characters. Math equations are often doubly and triply sub- or superscripted, they include some graphic elements, and they are the most likely candidates for typographic error. They represent a specialized subset of most technical typesetting, but such an important one that some efforts have been made—such as TeX™—to provide solutions for these difficulties. Having corrections in "hard copy"—that is, on paper—is more reliable. Later, if an error occurs in print, how it happened (and whose fault it was) is easily substantiated by written corrections. Sending hard copy galleys is better for both the publication and the contributor, though, primarily because any coded, transmitted text is harder to read. A lot of errors can creep in on the way to the press, and they may be errors that only the original author could recognize. Most readers can overlook a simple typographical error, but incorrect equations, garbled engineering specifications, and inaccurate scientific data will distort the reader's comprehension.

On galley proofs, a good editor will work with author corrections to keep the text intact and to meet deadlines. Editors also try to keep from having the galleys reset and sent out to the author a second time. A preliminary revision transmitted to the author could save a second set of galleys. The author would make his changes and transmit back, clearing the way for a good set of first galleys. However, the editor is unlikely to notice authorial changes in transmitted copy unless they are marked in an emphatic manner, and she will have to read the entire text again before transmitting to the typesetter.

In any editorial production process, the sooner errors are found, the cheaper it is to correct them. The cheapest time is before the author hands over the original manuscript to the editor. Later, correcting errors on galleys will be cheaper than correcting page proofs or blue lines. Corrections made during press runs cost more than they are worth, and they are rarely allowed by editors or printers. Thus, the timing and the depth of any edit save both cost and production time.

With electronic editing, some inexpensive correcting can be performed before the text is transmitted to the editor and before galleys are set on the phototypesetter. Having an editor, or a secretary with good language skills proofread the soft copy before it is transmitted improves the accuracy of typeset copy. It may even be possible to eliminate galley and blueline proofing.

Word Processing Departments

The next best thing to author originated keystrokes is to have a word processing department type in each manuscript and transmit the soft copy to the equipment the editors use. Again, the communications are not difficult.

The copy doesn't have to be perfect, but the keyboarders ought to use a spelling checker if they have one. A spelling checker merely compares words with a list of correctly spelled words, but it can catch a lot of typographical errors (for more innovative uses of spelling checkers see the chapter by Robert Krull). The keyboarders should also try to type the equations but will probably skip them and insert the word "EQUATION" instead.

Optical Scanners

What we used to call "OCRs" for "optical character recognition," scanners make good theoretical sense, but in practice a scanner is only another piece of equipment that someone has to learn to use. Like everything else, the technology has improved in the last few years, and the prices have become more reasonable. Scanners will probably be used more frequently in the future than they are now as they become more sophisticated, faster, and less expensive.

TEAM EDITING

Team editing sets up three demands. First, one editor must retain control over any changes made to a specific text; the editor-in-chief holds that one editor responsible for this text. Second, the proofreader or second editor must be able to make changes quickly and easily. Third, the computer must be able to keep up with all the changes.

All three requirements can be solved by word-processing software. Some of this software allows a writer to highlight or list out any changes made to a document during an editing session. If the software highlights change, the words may be displayed in bright characters or marked with a special "background" character—a non-alphabetic character or a punctuation mark. A listing feature usually creates a modifications file showing all revised and new lines.

A listing feature is not available with all word-processing software, probably because it is relatively unimportant in a single-user environment. When working without this feature, editors often agree among themselves that they will make corrections by using a particular typographic convention, i.e., all capital or italic characters, and they will make a copy of the original for reference. Beyond this kind of basic editorial agreement, a program could be written to compare the original file with any revisions and indicate any differences with a set of specialized highlighting techniques.

With a listing feature or with agreed-upon typographic revision signals, the proofreader examines the edited document and makes changes based on a knowledge of the special revision tracking techniques used by the editors. Later, the editor who is responsible for the manuscript reviews the proofed document. In this second pass the editor, who has a system with a listing feature, may only examine the modifications file. If the editor trusts the proofreader's judgment, this proofed file can now be electronically transmitted and typeset. But the editor always has the option to accept or reject any corrections.

Return on Investment

Team editing builds confidence among editors because it provides an opportunity to debate stylistic issues. It also provides a series of checks on developing texts that insure consistency and excellence. Texts that have been edited electronically have the added advantage that they are easily transmitted electronically to proofreaders and typesetters. The significant value of electronic proofreading and typesetting lies in the fact that these operations are performed at the level of the embedded generic type codes. Capturing errors at this level makes typesetting faster and more accurate.

EDITORS, TYPESETTERS, AND CODING METHODS

Editors are not typesetters. Whether the typesetting machine is across town or just down the hall, typesetting requires a lot of codes, or cryptic instructions, to specify font, point size, line length, primary leading, secondary leading, italics, bold face, superiors, inferiors, margin justification, paragraph indents, bullets, tabular information structure, and the like. Editors don't care about these machine instructions; typesetters do.

Before word processing began to change the ways words are prepared for publication, editors did not want to write out in longhand "9-point Century Schoolbook on 10 points primary leading, by 20 picas, ragged right." So they developed typesetting codes. They jotted these typesetting codes in the margins of typewritten manuscripts, and the typesetter posted a chart for the typesetting machine operators to go by. So an editor's arbitrary "D16" became the code for a figure caption and could have meant "8 on 9 Garamond bold italic, all caps, centered in 13.5 picas."

These codes depended on an annotated paper manuscript that had to be interpreted by a typesetter. When electronic text began to be transmitted over communication lines, these typesetting codes also had to be transmitted since the annotations were no longer available. Now the editor places "D16" in front of any figure captions, and the code is transmitted as part of the figure. The "D16" is eventually translated into instructions to the typesetting machine.

In some instances, however, typesetters may need more fully developed mnemonics, such as "/h$DA:A4" instead of "D16." While this is an extreme

example, it does point out the levels of interaction that can be required to produce a typeset text. The amount of keyboarding required for such an operation is often more than the author is willing to provide because it makes authors into typesetters rather than authors. Fortunately, most complex mnemonics of this type are now provided for or translated by the typesetter's electronic equipment.

In a sophisticated electronic editing operation, we want to perform this translation before the text file reaches the typesetter. This can be accomplished by an "intelligent," or discriminating, find-and-change program that keeps track of font size and style. It will be written specifically for the editorial department, preferably in an easily accessible language. Some "elegant" word processing systems provide "style sheets" that allow writers to specify the type of text to be entered, and the system enters the corresponding formatting codes automatically as the author types.

Most of the sophisticated coding and string matching would be too difficult for basic electronic find-and-change routines common to word processors. The translations are too complicated and too time-consuming. If the mnemonic for captions was coded with a "D16," for instance, we would only have to use find-and-change once to replace our string with the typesetter's code—"$LL1306$SZ08$STb4$PL09CCZL," but only if the figures were all the same size and all the captions were of the same length. If we had seven captions, we would have to use find-and-change for each caption code and be certain that we were careful to detect subtle variations in the coding that had the potential for confusing our translation scheme. Besides caption codes, we would probably have codes from D1 for a major heading to D20 for the author's acknowledgments.

To further confound the issue, suppose the font size for the subtitle is 18 points but is 9 points for the literature references. A superior (or superscript) in the subtitle would probably be 15 points, while one in the literature section may be 7 points. If our code to start superiors is "$S," we could not find-and-change "$S" uniformly throughout the document. We couldn't because we could not replace all of them with either "$SZ15" or "$SZ07" to indicate the proper point size. Nor could we run "$SX" through such a one-time "global" change because we would have to choose to replace each "$SX" with either "$SZ18" or $SZ09," and we have more text than subtitles and literature references anyway. With a find-and-change feature that lets us repeat a search for each occurrence, it will take us hours and we'll make careless errors, no matter if we know all the fonts and font sizes from D1 to D20 by heart. If the word processing system does not offer style sheets or some easy translation method, the solution is to use a computer program.

Generic Codes

Why should we do any conversion at all? One reason is to take advantage of "generic codes." You may have heard of GENCODE, a standard, generalized

markup language adopted by the Department of Defense. GENCODE is overwhelmingly complex and is not what I mean by "generic codes."

As an example of possible generic codes, let's say an editor is working on several book-length manuscripts. She knows that one week before each one must be mailed to the printer, the publisher will look at the sections she has finished and decide to change the format for the whole book, including one or more of the type fonts. In each book, the type code she has used for figure captions is a "/16" in her electronic text instead of the typesetter's "/h$DA:A4." If the "A4" part of the typesetter's code designated a particular size, leading, and line length in Helvetica bold, then a "B2" in the same position may indicate a size, leading, and line length in Century Schoolbook italic. For the second and third books, the publisher suddenly realizes that the Century Schoolbook italic will look infinitely better.

If she had used "/h$DA:A4," then this editor would not have to search for each occurrence and replace it with "/h$DA:B2" through some 137 electronic pages in twelve separate files for each of two books; she would have to search specifically for each use of "/h$DA:A4" in a caption. But there is a better way.

Because she has used a generic "/16" code for captions instead, our editor would now like to just call the typesetter and say, "The sixteen code for Books 2 and 3 is now Century Schoolbook italic. For Book 1, it is still Helvetica bold." The trouble is that the typesetter is unable to accept "/16" and still needs the entire "/h$DA:A4" or "/h$DA:B2" code. So instead of calling the typesetter, the editor tells her own computer program to make the change. The program is designed to make this easy for her, and it replaces each "/16" in Book 1 with a "/h$DA:A4" and each "/16" in Books 2 and 3 with a "/h$DA:B2." The style sheet functions of the newer word processors provide essentially the same operation. By allowing the user to redefine a style, the document or a subsection of a document will be formatted whenever a style is applied. The main difference between the two processes is that most style sheets reset the text without requiring the author to enter explicit formatting codes. But these style sheet codes will not be useful if they cannot be transferred to the typesetting machine, whether laser or phototypesetter.

More fundamental still, it takes fewer keystrokes with generic codes to type "/16" than it does to type "/h$DA:A4"—ten, counting the five shifts for the dollar mark, capital D, colon, and capital A's. Let the computer do all of those keystrokes with fewer mistakes than you can potentially make. A few changes in the program and the editors can start using "—16" instead of "/16" if that's what they prefer.

Finally, if the text is being set by various typesetters, generic codes can make it easy to change typesetters. The transmitted codes will probably have to be different for each typesetter, so let them be different for the computer, not for you. You can still use "/16" for captions, for instance, no matter which typesetter you use. In addition, each typesetter will probably tell you that the same

generic scheme can be accomplished through the use of "datas" or translation tables. The data code "/h$DA:A4" could as easily be "/h$DA:A16" to be used only for captions, for instance. As long as you rely on the same typesetter, you can use that typesetter's translation table, but again it will be different for each typesetting house.

Controlling Typesetting

To take the use of a computer translation program one step further, if the editor's computer translates every code anyway, why not substitute the typesetter's machine codes directly? In effect, the editor, through the computer, would then control the typesetting machine.

To an editor, this means that typesetters, whether in-house or across town, never get confused when the entire book is reformatted and all of the type codes are changed. In fact, the typesetters do not even have to know what changes have been made. For that matter, neither do all the editors. As long as one editor is in charge of the program and no one else tampers with it, changing the type font for a caption, for instance, becomes a matter of house style.

Alternatively, low-budget laser typesetting provides enough quality for publications like newsletters, seminar proceedings, and in-house computer documentation. A company may produce its own finished copy rather than sending signals to a typesetter. In this case, type codes will take a different form, but compatibility is again resolved by the WYSIWYG translation program resident in the system rather than by slower, less accurate, and more expensive human labor.

TRANSMITTING TEXT

If you are not sending electronic copy over the phone to an independent typesetter across town or through dedicated computer cables to an in-house machine down the hall, you are getting only half the value out of electronic editing. As long as editors mark up printouts, someone has to re-key the entire manuscript, inadvertently introducing errors. An editor once said, "If there are two ways to interpret an editor's instructions, typesetters can be relied on to choose the wrong one." That may be unfair, but a typesetter cannot be expected to interpret what the editor intended in a scientific equation, for instance.

Conversely, most editors have never been in front of a typesetting keyboard. They have no idea how difficult it can be to envision what the typeset copy will look like by gazing at the typesetting screen (much less the editor's hand-marked typescript). Viewing screens that narrow the differences are becoming more popular with typesetters—and they help—but the viewing screen is not the screen directly in front of the operator, and it doesn't look the same as the phototypeset copy will anyhow. Until you transmit text electronically, you have every reason to expect what you might call "a significant number of errors"—and to have to correct every last one of them yourself. And you'll be paying prime rates for typesetting.

Transmission Protocols

Each character in asynchronous communication is placed in its own package, so to speak, and is transmitted from one computer to the other as single characters. In asynchronous communications, a single character requires 10 bits: 8 bits for the character and 1 start bit and 1 stop bit to define it as a character. So a 300-baud modem (300 bps) takes one second to send 30 characters and two minutes to send a single-spaced page of 60 lines with 60 characters in each line. It is easy to calculate the savings in time provided by faster modem speeds such as 2400 bps.

With synchronous communications, the software packages in each computer ascertain where the characters start and stop when transmission begins and keep track of characters as they are transferred in large data blocks rather than as distinct characters. Synchronous communications is both faster and more accurate. Of course, speed and accuracy are balanced against the cost of personnel time and labor and the capital costs of purchasing equipment. In the future, with telecommunications based on lasers and fiber optics, transmission speeds in the neighborhood of 10,000,000,000 bps will be taken for granted, and the computers will be running on crystals and light diffraction.

This chapter has provided an overview of the ways in which text is placed into a computer-based word processor and how that text is massaged by editors to provide useable signals for typesetters. It has also offered a brief glimpse into the area of text transmission over telephone lines. All of these functions have as their primary purpose the preparation of a text that is amenable to electronic alteration throughout the early part of the document development cycle. They provide the means for working with text that are versatile and flexible and represent a distinct departure from traditional techniques used in hand editing.

CHAPTER 10

Word Processing and Electronic Publishing

PHILIP RUBENS

When using a word processor writers are concerned primarily with text creation. The computer, with its ability to easily store and manipulate signals, allows writers to type relatively "clean" texts without having to produce multiple paper drafts. More sophisticated word processors allow formatting, usually with embedded commands; some typeface selection, text spacing, and the like; and a few basic graphic abilities for creating charts, graphs, and tables. Few word processors allow significant graphic manipulations or simultaneous text and graphics incorporation into one document. As a consequence, we can envision a word processor as producing a single file that has text and some basic format commands in it.

Creating a text remains the writer's responsibility. Figure 29 depicts the typical word processing working environment. The writer inputs characters in much the same way that he would at a typewriter. Standard page definitions may also be placed in the text to create specific formats. While the writer often performs

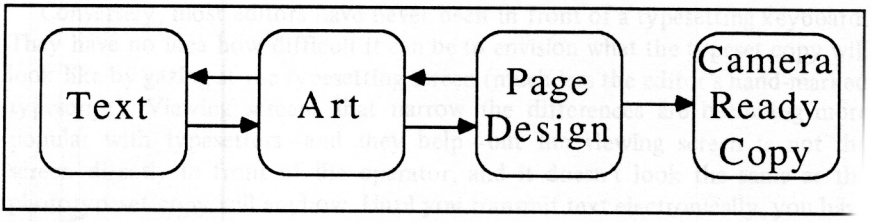


Figure 29. Word processing workflow.

this task, a production editor will probably check coding accuracy before printing the final copy. The hardcopy—final—output from a word processor can only be as good or bad as the available printing device linked to the word processor. Typeset or book quality can be achieved on a photo composition machine, while dot matrix printers usually provide only draft quality.

Regardless of the output device, the final product will still look like traditional "galleys," long pages of typeset text that must be pasted into layout sheets with accompanying artwork (anything more sophisticated than charts, graphs, and tables, at present). Added to these galleys will be finished artwork, usually created on stand-alone electronic equipment, by camera, or by hand. Rules, lines, decorative borders, and the like will be added by keyline artists. Finally, prepress artists will photograph these layout sheets to produce printing press plates. Other than the word processed text file, this process is identical to traditional publishing.

Rendering a word processed text, prepared by a writer, into a finished publication, then, requires additional treatment and handling by graphic artists, keyline and paste-up personnel, production editors, and printers as well as further discussions among writers, artists, and printers to resolve any substantive differences about design and useability. The results of these discussions generally require altering the text file or changing the relationship of page elements. In either case, a word-processing system does *not* provide a means for all participants to work on the same text either electronically or simultaneously. Thus, it is not uncommon for the paste-up process to be repeated several times based on group consultations during the development cycle. The typical process for creating a text on a word processor depends on a great deal of external treatment, multiple copies, and group consensus before it can be printed. Furthermore, the process provides varying levels of support and interaction for the participants. Typically, they work in isolation on equipment that varies from electronics to pencil and paper. This description should not be taken as a contradiction of the capabilities outlined by Patricia Caernarven-Smith in an earlier chapter. What she aptly describes is a conceptual framework and methodology that obviates many of the problems considered in this chapter. My concern here is with the typical uses of word processing and not the "elegant."

In contrast to the single file of the typical word processor, we can think of a desktop publishing system (Figure 30) as a series of three related files—text, graphics, and format—each dedicated to one of the major steps in the production process and able to be integrated electronically and simultaneously. Writers type signals (words) into a WYSIWYG (What You See Is What You Get) system. That is, she sees exactly, within the screen's limits, what the final product will look like. At the same time, artists or other writers manipulate images to combine electronically with the text. At any point during the document development cycle the text and graphics files can be merged into a format file that allows all participants to see the final product. Since the final product can be displayed

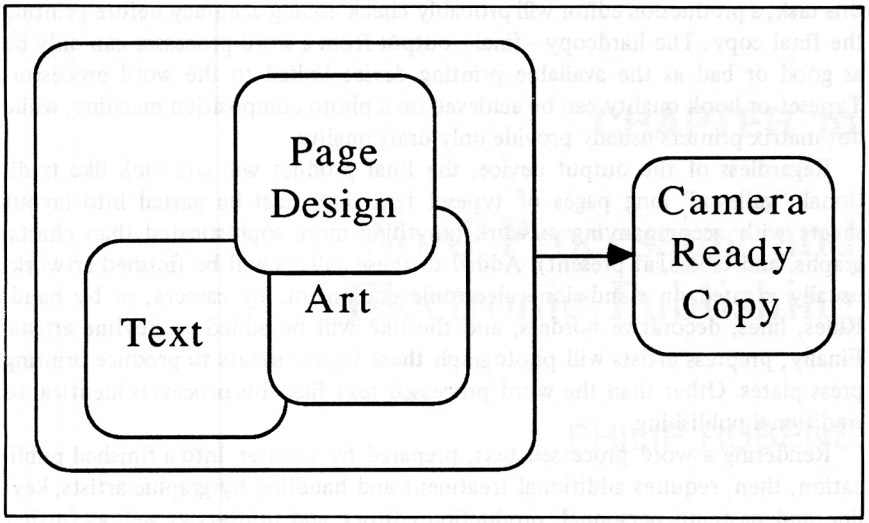


Figure 30. Electronic publishing workflow.

electronically, it can also be altered on the display screen until a group design consensus is reached.

The primary advantage of electronic publishing is clearly evident in the level and amount of interaction that it provides. The repetitive process of preparing hardcopy (paper) layouts and arguing design considerations is all accomplished electronically. Thus, final copy (analogous to the text galley found in both word processing and traditional publishing) will often be camera-ready. On very sophisticated systems, the entire process occurs electronically, even the preparation of negatives for printing plates. Word processing's repetitive and isolated tasks are eliminated by electronic publishing because *all* design work occurs on the same medium simultaneously. This is, indeed, an exciting prospect; one that I would like to explore in this chapter.

CAPABILITIES OF ELECTRONIC PUBLISHING

Electronic publishing addresses the need for faster creation and production of the tremendous volume of business and industrial information. While newspapers and magazines first implemented some form of electronic publishing, the current emphasis is on creating equipment to support technical publishing. This market accounted for about 2.5 trillion pages of documentation in 1984, according to Xerox Systems Group. That staggering figure, however, pales beside the fact that most technical pages go through slightly more than a dozen revisions during their life cycle.

Corporations devote from 6 to 10 percent of their budgets to producing paper communication. It has been suggested that desktop publishing systems can reduce production time by nearly 70 percent and costs by 30 to 50 percent. Indeed, one publication, a national news magazine, reduced its costs by over 22 percent in five years, even though it was investing heavily in new production equipment during that time. Similar savings have been reported by other high volume, text-intensive operations such as newspaper, catalog, and magazine publishers. Current costs for electronic publishing systems are about one dollar per typeset page compared with twenty to thirty dollars per page for traditional publishing. Thus, it is obvious that electronic publishing has a potential market in technical publications and it has the potential for answering the need for timely information produced within a conservative budget. Like its predecessor, word processing, electronic publishing introduces new tools for writers and graphic artists. Productivity, how long it takes to produce a document, can be influenced in very positive ways by these tools.

Traditional Word Processing

The new technology expands on the way a typical word processor operates. We type characters on a keyboard and the word processor stores them on electronic media, usually a disk. Most word processors allow some variety in page formatting. Usually this formatting uses either embedded commands or WYSIWYG. The word processor I am using depends on embedded commands to format a printed page and I must imagine how the final page will look. In contrast, most electronic publishing systems are WYSIWYG. That is, the writer selects type faces, sizes, and styles while writing and actually sees them on the display screen. Except for differences in the resolution of the displayed and printed image—displays usually have 70-100 dots (or picture elements "pixels") per inch, while typesetters range from 300-2400 dots per inch—the appearance of the display screen and the final printed page are remarkably similar. However, because of its lower resolution the screen looks "fuzzier."

WYSIWYG offers a distinct advantage to the writer in that she sees how the final text will appear in fairly complete detail. One does not want to give up some of the more useful features of word processing to gain these advantages. Hence, most of the more useful word processing features are still available. For instance, you can create tabs (much like a typewriter); you can add special format features to control lists, tables, indents, and the like. You can also check spelling and grammar. Thus, electronic publishing offers many of the features that we have already become familiar with in word processing and adds several tools to make the publishing task even easier.

This, however, is still a narrow picture of electronic publishing's potential. It is more than simply adding features to traditional word processing; it is also more than offering writers pretty pictures on a screen. Electronic publishing

combines three kinds of files—text, graphics, and format—while providing the tools to support the traditional tasks found in publishing houses—writer/editor, graphic artist, and production personnel.

Since electronic publishing integrates these tasks into one environment and medium, it is tempting to imagine a publication group in which all three tasks can be performed by a single individual. While that is the ideal conception of the electronic publishing system, it is far from a reality. One of the tasks of the remainder of this chapter will be to suggest ways in which this concept can be realized *and* to point out the limitations of such a vision.

Electronic Art

In the past, fine art created on a computer was usually limited to all but the most expensive and sophisticated machines. Today, virtually all electronic publishing systems provide methods for creating and manipulating art.

At the most rudimentary level you can select from "canned" or "click" art that is prepackaged, stored on a disk, and electronically manipulable. To use this art you either allot whitespace, an open area in the text, or simply select the location for the artwork. In either case you retrieve the artwork from storage and electronically paste it into place. The major difference is that the first method requires you to determine how much space to reserve and in the latter the system makes the determination.

Such art can also be altered in a few, highly controlled ways: adding texture, providing text, and the like. More sophisticated publishing systems provide the capability to create art from scratch, either on the screen with electronic art tools such as a graphics tablet, or to enter existing art into the computer with scanners, electronic digitizers that "read" pictures or scenes. William Benzon provides a sample of the ways in which a picture can be both scanned and manipulated in Chapter 8. Regardless of the input device or medium, the artist, and the design-literate writer, now have the ability to create original art that can be easily altered during the design process. This art is also available to writers and editors who want to combine it with their texts.

Electronic Typography

One of electronic publishing's newer capabilities involves the manipulation of type. In the past, writers using a word processor had little choice about type *and* had to wait until the page was printed before they could see how their type selections looked. Typically, word processors allow mixed case (upper and lower case type), boldface, and italics. The typeface, as well as access to *any* typographic variation depends on the printer's capabilities. "Daisy" wheels, for instance, are highly restrictive since you have to physically change wheels to access a type variation. Dot matrix printers are slightly more flexible, although their type size and font range are restricted.

Electronic publishing provides type variations commonly found in traditional typesetting (although with some limits), the ability to alter or create new type faces, and the capability of combining these creations in novel ways. One can, for instance, position a typical typeface on a page, then manipulate that type by "bending" it through a series of angles. Such capabilities have significant potential for cover design and marketing literature. All of these artistic effects can be seen on the display screen, can be changed at any time, and can be used with a variety of programmable printers.

Electronic Layout

The final result of all these manipulations is a readable page. That is, the electronic publishing system must be able to combine texts, charts, graphs, and other graphics, with appropriate type selection, and layout features—column widths, page depths, picture placement, and the like—to produce a finished layout.

The difference between electronic layout and cut-and-paste are significant, perhaps even more than the difference between the typewriter and the word processor. In traditional publishing, even with word processing, tasks are divided into discrete responsibilities. The writer writes, the artist produces visuals, and the layout person integrates their work (Figure 29). The process is characterized by isolated production on dissimilar equipment; considerable discussion, evaluation, and compromise; and multiple layouts. While it is difficult, and perhaps undesirable, to eliminate the human interaction necessary in publishing, it is possible to alleviate its repetitious quality. Patricia Caernarven-Smith's chapter does much to describe the decisions and interactions that go into creating a word-processed text; the key differences between that process and electronic publishing are the ways in which the technology reduces the paperwork, provides interactive methods for incorporating texts, graphics, and page design, and allows the participants to focus on the total *finished* product.

Electronic Production

Once the page layout has been established, then, it is possible to print a copy of the final page. In small publication groups or writing "shops" this is usually done on a laser printer that has a resolution of 300 dots per inch. To the uneducated eye this type looks remarkably like a typical book or newspaper; in fact, most of us have been reading laserographic type in our newspapers for some time without noticing it. Slightly larger shops have access to a photographic typesetter with a resolution of about 2500 lines per inch; this is equivalent to high—textbook—quality type. It is virtually impossible for the average reader to distinguish these documents from typical texts.

High resolution systems also create excellent quality halftones and artwork. The basic artwork for Figures 29 and 30 in this chapter typifies the line art one can create electronically with even the crudest of printers.

Unlike line art, photographs must be made into "halftones" before they can be printed because a press only prints color or no color. If a photo is not halftoned, a press would "see" its image as a single printable (and inkable) character and would apply ink to its entire surface. Thus, any picture would be printed as a black block. The halftone is created by a "screen" placed between a photograph and a light sensitive medium. By exposing this combination to light, the continuous tones (white to black through a series of greys) of the photo are reduced to groupings of dots—some close, some large, etc. By breaking up a photo's image into dots we, in effect, "trick" the press into "seeing" the photo's image area as a field of dots and putting ink on only those dots. The perception of the printed photo relies on our ability to integrate these printed dots as continuous tones. Thus, big dots close together look black; small dots far apart look dull grey. Many electronic publishing programs, printers, and phototypesetters allow halftoning.

Electronic publishing systems also provide additional photographic treatments such as specialized screens, high contrast photographs, posterization, and direct negatives. These photographic possibilities can increase the artist's electronic range and reduce the production cycle by eliminating traditional photographic steps.

Electronic Page Design

One area in which this new technology can make a significant contribution is in unique page design. Creating page designs requires some degree of design sophistication and programming ability. Such programs depend on the designer's ability to learn a page definition programming language. Obviously, it also depends on that person's ability to imagine innovative and useful page designs. To use this feature an artist and/or writer develops a page design in paper and pencil then she creates a program that emulates that design electronically. The advantages in this system are 1) reprogramming design changes can be made relatively easily, 2) a library of basic designs can be developed and altered on a per client basis, and 3) a library of default (typical or standard) page layouts can be developed. This capability should prove especially useful in cover design, advertising, marketing applications, and technical publications where standardization is necessary.

Page definition languages also afford the artist the opportunity to add color to designs. While color and color separations are not available on most equipment, they have made some inroads at the higher end of the market. In magazine publishing, for instance, a number of systems provide color separations. They also allow the artist to manipulate colors and picture features. These are WYSIWYG and, unfortunately, require a great deal of computer memory. For example, digitizing a black and white photograph (continuous tone photograph) requires about a megabyte of memory. The same photograph, in color, requires about twenty megabytes.

SHORTCOMINGS OF ELECTRONIC PUBLISHING

These innovations all sound very exciting and "Buck Roger-ish" but there are drawbacks to implementing this new technology. For instance, while Patricia Caernarven-Smith makes a compelling and justifiable case for the abilities of the writer in her chapter, there is justifiable concern about the graphic abilities of the typical writer or editor. The major task of the writer and editor is text production; as a consequence, they are often oblivious to the fine details of final production, the steps necessary to get a text into a useable "form" in a reader's hands. Given this general lack of background, it is unlikely that untrained writers and editors will be able to assume the roles traditionally reserved for production personnel without a significant training period, the consulting services of production personnel, or an in-house production staff (who will also likely need additional training). This does not mean that the roles of writer/editor/artist could not eventually be assumed by one individual; instead, it suggests that we should not consider this new technology solely as a method for reducing personnel. For at least the near term, all major roles in the text production process are still necessary.

There are also still limitations on the capabilities of the production machinery. Laser-based typography is *not* typeset quality, although its quality is rapidly improving. Typically, laserographic typesetters have templates for producing one size of any given typeface. To produce larger or smaller versions these typesetting systems simply enlarge or reduce this template. Unfortunately, this practice destroys the balance, weight, and tone of the type face, making it less legible. Other, more technical, objections, such as the lack of kerning capabilities (moving letters closer together), to laserographic type are rapidly evaporating as the technology becomes more sophisticated.

Writer/Editor Role

In traditional writing groups the roles of writers and editors are carefully defined. The writer generates text and the editor makes the text useful for a specific audience and purpose. While the final decision for the format of that text usually resides with a graphic artist and layout person, the writer and editor have some input into the process. Patricia Caernarven-Smith discusses how writers can influence the design process in a word processing system. They probably have more input in word processing because they generally provide copy in "near" camera ready form. Their ability to do so reduces the paste-up process.

Writers and editors are usually asked to advise on type size for headings and the placement of location devices—headers, footers, page numbers, section headings, and the like. For the most part this advice is fine-tuned by the art staff. The artists know, through practice and education, what effects on readers can be created by the grosser features of texts—page size, type size, leading, picture placement—as well as the more subtle features of type style, color, specialized screens for illustrations, and the like.

Everyone, however, knows that layout decisions represent only the outward and most visible expression of the basic organizational principles underlying the text. Thus, if page designs are intended for a reference manual, then the text must be written to aid retrievability and comprehension and the layout must provide easy identification of key elements. Page designs and writing standards *must* work together to achieve the intended purpose for a specific audience. Once basic design decisions have been incorporated into guidelines, the publication process becomes standard.

Artist's Role

The traditional role of the artist in the publication process has been to prepare artwork and to make decisions about page and book design. This means that the artist has the responsibility for selecting and modifying photographs. They are also responsible for creating artwork such as tables, charts, graphs, line art, and the like. While electronic publishing affords the untrained writer the opportunity to enhance a text with digitized electronic art, the artist, as William Benzon points out in Chapter 8, is more capable of manipulating this artwork and incorporating it into a finished text.

For example, a simple task such as selecting appropriate line weights for boxes or tables can demonstrate that the artist's eye is still necessary in using the electronic medium. Line weights typically range from hairlines to 12 points in thickness. For most laserographic processes, reduced character resolution makes hairline rules illegible, while lines above four points tend to "overpower" the text they contain. Writers would not know this and would specify tables with line weights above four points; artists would not do so. Arguably, writers both will and can learn these rules. Sometime in the future, then, basic art decisions may become simple.

Another, and perhaps more telling, example involves the picture quality possible on typical laserographic output devices. Photos, you will recall, become printable by filtering them through halftone screens that range from very coarse (about 80 lines per inch found in newspaper photos) to very fine (300 lines per inch used in high fashion photography). In contrast, a laserographic process depends on picture elements (pixels) and has a resolution of about 300 pixels per inch. This is a decidedly different process from halftone screens (lines versus pixels) and requires a specialized knowledge of the measurement systems, the relationship between the systems, and an ability to visualize the effects of translations between the two media and processes. While the precise relationship is beyond the scope of this paper, it should be enough to point out that a 300 pixel per inch output device will not have the same resolution as a 300-line halftone screen. The process is further complicated when production calls for the use of specialized screens; all of which can now be created on even the most basic phototypesetters.

Production Editor's Role

Like the artist, the production editor makes decisions about the form and production of the printed text. Many of these decisions have to do with the possibilities and limitations of the printing process. For instance, the printer may require that texts be prepared in particular ways in the pre-production stages to optimize their equipment. The production editor decides whether text will be "ganged" into particular negatives, whether "work-and-tumble" or "work-and-turn" will be the printing process, whether screened images will be printed mono- or duo-tone, and the like.

I was deliberately technical in the preceding paragraph to demonstrate something of the specialized knowledge required to produce texts. Let me be a little more specific about some of that knowledge to demonstrate its importance in the production process. Most word processing and electronic, in this case—desktop, publishing is prepared on a small screen; usually the writer or artist can only

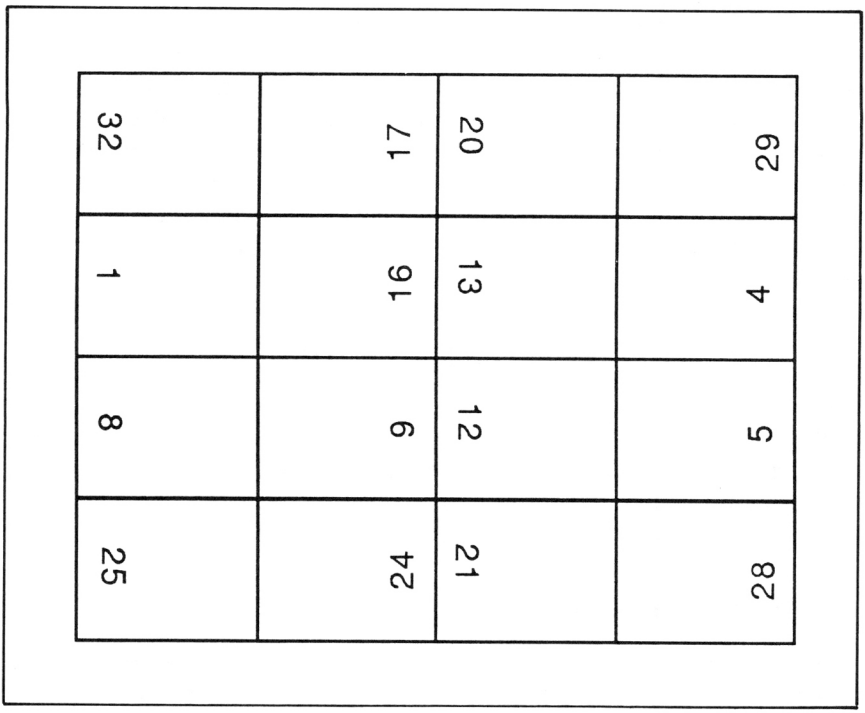


Figure 31. Typical imposition for A0 sheet (approximately 35- X 45-inch paper, exclusive of printer margins).

see part of an 8½- X 11-inch page, though some can see an 11- X 17-inch page. The production editor must make decisions about page placement (imposition) to optimize the capacity of the presses. For instance, if a press can print on uncoated offset paper, that paper is available in a standard 35- X 45-inch sheet which can be cut to 22½ X 35 or 17½ X 22½ inches. The standard sheet can print sixteen pages per side, the second size eight, and the third four. Based on the press capacity, the production editor will place the requisite number of pages in the "correct" order and orientation so the finished sheet can be folded and cut with the pages in the proper sequence. Consider Figure 31 to get an idea of the complexity of such a typical production project.

What I posited in the previous paragraph was a fairly simple production problem. It could be further complicated by the necessity of adding a second color to *only one side* of the printed sheet. This reduces printing costs and is a *very* typical consideration when planning a print run. Obviously, successfully planning for such production need is no small accomplishment when one considers the typical many-hundred page technical manual. The other problems I alluded to earlier necessitate similar production knowledge. None of this knowledge can be learned by simply using electronic publishing equipment.

Given the nature of electronic publishing systems—the ease with which its basic tools integrate the writing, graphics, and pre-production processes—one is left with the impression that the production process will be trivial and will somehow, miraculously, be handled by the writer sitting at an individual workstation. Nothing could be less true. At the conclusion of the writing process, after the graphics have been incorporated, the camera ready copy must still be treated photographically and stripped into a carrier medium to make printing plates for a press. That is a complex operation and requires a significant body of graphic knowledge.

THE IMPENDING FUTURE

I started this chapter by announcing that basic word processing was being changed by the introduction of electronic publishing workstations. Even as I write I am aware that the existing technology I have described is well on its way to obsolescence. Let me, for a few moments, describe some innovations emerging from research centers or available in super systems.

Improved Storage Devices

Typically, electronic signals have been stored on computer disks or tape. These media can store an enormous amount of data in a very small package; even 3½-inch disks store over 100 typed pages. Other media have boosted this storage appreciably. For instance, compact laser disks can store thousands of pages, although at present they can only be written once and cannot be altered. While

this technology is applicable to storing images that are not intended to be modified, it will not be long before they allow over-writing and become as versatile as the floppy disk. Even with their present limits they are useful for storing art images that can be read into memory and used with text in other documents.

It is also likely that new input devices will increase the usefulness of electronic publishing systems. As William Benzon points out, most digitizers scan pictures very slowly. The most advanced technologies can digitize both color and black-and-white pictures very quickly and establish a digital "value" for all parts of the picture, much like taking a light meter reading of the picture and recording it for later comparisons. With these systems it is possible to ascertain and alter the value of any single pixel of an image; it is also possible to create a printing plate directly from this digitized image without further mechanical or photographic darkroom treatment. This pre-production control of images can be enhanced by the computational abilities of some presses that already monitor color and image density to control quality during the printing process. Once new input capabilities are combined with the actual production process the level of image control will be significant.

Beyond simply allowing for more efficient and qualitatively better printing, better digital input devices will provide for further tonal discrimination and the possibility of on-screen color separations. Some highly advanced systems now offer such capabilities. The versatility of these systems even allows the production person to impose color separations directly onto a printing plate without an intermediary process. It is probable that the technologies that drive these systems will filter down into the desktop publishing market in the future.

Improved Communication Capability

At the upper end, electronic publishing systems are generally stand-alone workstations; communication links specific workstations to peripheral devices for input and output. Thus, the writer, the artist, and the production person all share responsibilities on a single project even though they are working on separate machines. The typical small electronic workstation can be tied together in Local Area Networks (LAN), can be independently tied to input and output devices through external ports on each machine, and can also have access to larger telecommunication networks that may stretch beyond the immediate workplace.

At present many of these capabilities are still somewhat primitive. That is, LANs can be easily overtaxed by demand and somewhat "slow;" telecommunication speeds depend on code translations that are not always successful; and peripherals do not always work on every machine configuration. All of these problems require consistency beyond the scope of any single manufacturer (and beyond the scope of this chapter to examine); it calls for a global agreement among manufacturers about the shape of codes and the protocols required to drive and understand a variety of machinery. Given the competitive nature of

computer sales, such an agreement is hardly likely. Thus, we must await better translators to make electronic signals truly, and accurately, portable.

Improved Typography and Layout

The biggest single complaint about electronic page composition is that the type is not "real" typography. That is, it does not emulate the stroke width to height ratios, it does not account for the various heights of small caps versus lower case letters, it does not have similar counter space, and it does not allow kerning (the ability to tuck one letter closer to another). I said earlier that even laserographic typography would be indistinguishable from typesetting to the typical reader. While I cannot prove that assertion, I have pointed out that most newspapers and magazines have been using a laserographic process for some time without deleterious effects on readers.

While it is still possible to decry the decline of typography, it is likely that some significant changes will occur in typographic design. Hermann Zapf, a noted type designer, recently recounted his futile experiences in first translating some of his type designs to film, for photo typesetting equipment, and then to digital signals. In both cases the media were not resilient enough to account for the artistic nuances of the typeface. Zapf maintains that it is the artistic quality of type that should be preserved; from the standpoint of type as an artistic expression that is decidedly as it should be. Zapf mediates this position by maintaining that digital, or non-metal types that do not require traditional hand forming, should be designed for specific media and needs. His efforts in creating the AMS-Euler font is a good case in point. This digital typeface was created for the American Mathematical Society and includes several hundred characters and symbols germane to writing about mathematics. Zapf created this face using *METAFONT*, a type design computer language created by Donald Knuth. Knuth has had his own adventures in type design using this language that are well worth considering.

OF HAND PRESSES AND LASERS

I have this nagging suspicion that much of what is in this chapter is already well on its way to obsolescence. But I want to consider for a moment the disjunction of the hand press and the laser. Interestingly they represent two extremes in publication technology and, equally, they are both still just technologies. Their function is to make life a little easier for the writers, graphic artists, and printers. I have been deliberately evaluative throughout this essay because I wanted to use the opportunity to point out that this new technology is only one moment, granted a very exciting moment, in a long line of development that stretches back to the hand press, even further to early scribes and paper production from papyrus. I wanted to focus our attention on the tasks as

much as on the technology. It seems possible that we can become too enamored of the bells and whistles and forget that there are some extremely subjective decisions—both in terms of writing and visuals—upon which the "technology" must depend.

In the early part of the chapter I detailed all of those functions germane to text production that electronic publishing improves. It provides a fairly close approximation of the finished page; it opens new avenues for creating visuals; it allows better control and design of type; and it facilitates basic page layout.

I countered that optimistic picture in the second part of the chapter by discussing some of the tasks electronic publishing doesn't do well. Most of these tasks require better training and integration on the part of the staff. These are very human issues that most technological innovation ignores. For instance, I argued that the traditional text production roles should be maintained. I have not taken this approach for purist reasons. I believe that simply using the technology will *not*, in and of itself, teach every participant enough about all of the facets of production to make certain personnel superfluous. I would suggest, in contrast, that maintaining the traditional roles has the potential to make the new technology even more viable and exciting. In discussing the impact of this technology with some of its technical specialists, I have been more than encouraged to learn that it is the artisans—both writers and artists—who are making a concerted effort to learn how these new devices operate. They are also making significant contributions by designing objects—pages and art—that are clearly beyond what the technical creators of the equipment thought possible.

The chapter ends by offering a glimpse into my own personal crystal ball. What I provided there was the briefest suggestion of some innovations that are currently in the planning and development stages. To assess their potential and their impact on text production would be inappropriate at this time; however, it is obvious that this line of development has struck a creative note and will create considerable interest in the graphic arts and text production industry.

While word processing by itself has helped with revisions, revision tracking, and production times, it still has a considerable "hand" composition element. That is, once the text and visuals have been created, they must be formatted, combined, and photographically treated before printing can occur. Electronic publishing allows most of this process to occur simultaneously; the end product can be camera ready copy. I have been fairly open about the shortcomings of this technology; I have also pointed out that many of these problems will be shortlived. Better, more efficient, easier to use, and "elegant" equipment appears almost daily. It is likely that the quality of these newer innovations will have an even greater impact on the communication and production process.

Epilogue

CHAPTER 11

Learning to Write with a Word Processor

ANNETTE BRADFORD

There is an age-old argument in linguistics that says only one of these statements can be true: "Language enables thought" or "Thought enables language." (This is the linguistic version of the chicken and the egg.) I want to propose that neither of these is correct. Thought, I maintain, is a motor skill.

I admit that this is a radical hypothesis, but my last six years of writing experience using a computer have proven to me that this is the case. In fact, I'm sure that since I began using the computer as an instrument for generating ideas, my brain has *moved*–from my head to my fingertips. It is as if I want to rewrite mythology to say that Athena sprang not from the head but from the hands of Zeus. The physiology of it is incredible. Perhaps this hypothesis seems more plausible if I explain to you that it didn't happen overnight. It was a gradual process. For me, the computer began as an instrument for typing papers, but has evolved into a medium for generating ideas. I say this to comfort you. I want you to learn to appreciate the virtues of the computer as a thought-generating tool, but I admit, if I awakened one morning to find my brain in my fingers, I'd be worried. Rest assured that the transition takes place over time.

In the pages that follow, I will not attempt to impress you with the *uniqueness* of my experience. I've listened to too many conference papers that say "Wow! All I did was put a ".p" here and I got a paragraph" to believe that my experience

with the computer is extraordinary. It is not the uniqueness of my experience that makes this chapter useful for you. Rather, I would like for you to read my chapter and then, in the years that follow, as you evolve as a computer user, mutter "yes . . . yes . . . yes" to yourself and think of me.

From the perspective of one with several years of experience teaching writing and several years experience with using a computer to write, one who has taught word processing, and one whose occupation is now writing and planning user manuals for computers, I want to outline the stages that I passed through in becoming a computer user. Then I will review some of the problems I encountered learning to use the computer, both from the perspective of a writer and of a teacher of writing. Finally, I will speculate on the potential the computer has for the writing teacher.

STAGES IN COMPUTER USE

Most people I know who have learned to love computers have gone through distinct and constantly evolving attitudes toward the machine. The stages I would define are:

- Fear and Loathing
- Use-It-Like-A-Typewriter
- Computer Composing
- My Brain Has Moved
- What Else Can It Do?

Fear and Loathing

Every humanist who is now a computer user has her own "fear and loathing" story. Mine began when I left Alabama in 1979 to pursue my Ph.D. in rhetoric and communication at Rensselaer Polytechnic Institute (RPI) in Troy, New York. RPI was one of only a handful of schools in the country at that time that offered an advanced degree whose object was to teach writing. I had barely recovered from moving to this yankee-infested part of the world when my new RPI advisor tried to sign me up for a computer course. Being someone for whom balancing a checkbook is a traumatic experience, I informed him, "I'm a humanist. I didn't come here to be turned into a computer nerd." That fit gave me a year's grace.

During that year, I watched my classmates turn out graduate school papers with incredible ease using the computer. My old manual typewriter paled in comparison. I discovered that I needed the computer much more than it needed me. When the chance to sign up for the introductory computer class came around again, I needed no prompting.

My introduction to the computer was a course called Computing in the Social Sciences. It was meant for graduate students in nontechnical curricula who

wanted to learn basic computing principles and applications. The course had historically been a "make-it-or-break-it" course for students entering my department's masters and doctoral programs. It was the best indicator of whether people with humanities backgrounds could indeed find happiness in technical writing careers. Most can, but some can't.

The professor who taught the course had developed an imaginary simple computer called the "Initiate's Digital Machine"—the IDM (no relation to the computer company with the similar name). The undergraduate computer hackers at RPI who graded our papers called it the "Idiot's Digital Machine" but we were undaunted. Using this simple model, we learned how real computers processed instructions and finally wrote quasi-machine language programs for the IDM. We learned basic Boolean logic and binary math. But before all this became too abstract, the professor did the one thing guaranteed to hook us: he taught us word processing.

After a couple of introductory sessions, I left class clasping my first word processing assignment. I had been dubbed "A97V" by computer services and assigned a password that had nothing to do with Allen Ludden. I sat down cautiously at a terminal and typed the sign-on phrase "signon a97v" with one finger to minimize the possibility of error. The screen flickered and THE SYSTEM responded "Enter Password." I typed my assigned string of nonsense characters and pressed the Enter key. I got the standard ready prompt. My existence was validated. I was part of THE SYSTEM.

So far, so good. I was ready to begin my assignment. I reached for a book that I had placed on top of the display above the keyboard, but I missed. The book fell and hit the keyboard with a thud. Awful things happened. The screen flashed white and filled with nonsense characters. Somewhere, I was sure, the new mainframe had just bitten the dust. And this was only the beginning of my "fear and loathing" stage. From these hardly auspicious beginnings, I developed a sense of what the computer would allow me to do. I began to type with all my fingers again. My dependence on the computer grew. I passed easily and quickly into the "Use-It-Like-A-Typewriter" stage.

Typewriter-Like

Armed with my new ally—word processing—I went forward to conquer graduate school with renewed vigor. But being a pencil bigot, I continued to write out all my papers by hand and used the computer only when I was ready to transcribe my handwritten pages. Correcting typos sure was easier when a clean copy was only a keypress away.

At this stage, I thought of the computer as a "super-typewriter," and the metaphor was a useful one for a while. It made the new and hostile machine seem more familiar. But ultimately, this way of thinking about the computer limited my thinking. For example, I under-used block moves and the massive

kinds of restructing for which there were no corollaries in the typewriter world. I had to rework my concept of the machine before I could pass comfortably to the next stage, Computer Composing.

Computer Composing

Once I realized the benefits the computer had over a typewriter, I began to use it for much more than processing finished papers. Rough papers that I had keyed into the computer got their finishing touches and their final paragraphs at the keyboard. I began to use my pencil less for writing than for sketching out the flow of ideas. To begin a new paper, I would copy my idea flow (which looked much like a sentence outline) onto the screen and then use these ideas as topic sentences or signposts by which I could check my progress and assure some kind of first draft continuity. Even at this level of use, I was astounded at how much more productive I could be.

This kind of expanded sentence outlining was the beginning of my computer composing. Unlike my husband (whose many years of journalism experience had made him comfortable with the typewriter keyboard as a composing tool), I had never been at home with the typewriter. Mistakes were too hard to correct (how many good ideas had I lost waiting for correction fluid to dry!). But even more troublesome is the fact that a typewriter displays only one line at a time—and that line is static and "finished" (in the sense that I can't back up to it several paragraphs later and change it easily). Another problem, even with modern electronic typewriters, is the absence of context. The unit of creation is the line, not the paragraph. I can deal with only one line of text at a time.

But the computer solves all these problems. With the display screen, I found a freedom from the tyranny of error and a medium for experimentation. Soon, I reached the point that I resented having to use a pencil. Letters to family and friends were all computer-generated. I began to feel itchy on airplanes or when I had time to kill if I didn't have a computer keyboard to help me pull my ideas together. In retrospect, I realize that my physiological transformation had already begun. I was locked onto the silicon path that can only end with brain relocation.

Brain Movement

I don't remember when I realized that I was no longer able to think without a keyboard. While I was still a student, I became so dependent upon the machine that I took my doctoral qualifying exams on the computer. I used it to develop materials for my classes and my consulting jobs. I am convinced that the fifteen months I spent writing my dissertation would have been at least doubled without computer text processing. The acknowledgments section of my dissertation contains fond words for the machine that had caused me such discomfort at first. In fact, after computer use at RPI doubled while I was a student and made it difficult to get access to a school terminal, I negotiated the only student loan of

my lengthy education to buy a home computer with a modem for communicating with the mainframe at school.

At this point in my transformation, I had the added incentive and pleasure of teaching other people to use the computer. Just like adults who relive the delights of childhood through their children, I could experience several times over the transformation of friends and school associates as they passed through the stages that I describe above.

I spent a long time at this stage. In fact, once your brain has completed its move to your fingers, you may never need to change the way you use the computer. If all you ever want to do with the computer is text processing, then finding a dependable and familiar program may be all you ever intend to do. But eventually, you may ask yourself, "If the computer improved my writing so dramatically, what else can it do?"

What Else Can It Do?

Even though I hadn't realized it, I had learned a lot more than text processing when I became a proficient user of both the large mainframe computer and of a personal computer. By learning a few programs in depth and learning to transfer data between the host and personal computer, I was creating for myself a set of computer concepts that I carried with me when I learned other software programs. I had learned, for example, that I could copy files from one diskette drive to another or rename files and store alternative versions of an idea on the same diskette. I had begun to acquire what software psychology expert Ben Shneiderman refers to as "semantic knowledge"—computer concepts that are system independent and learned by practice.[1] The opposite, Shneiderman explains, is "syntactic knowledge," system-dependent information learned by rehearsal (for example, the exact command for editing a file on a certain text processor).

Even though I had begun by learning other word processing programs, I decided that if text processing programs could make writing so much easier, what could the computer do for statistics? . . . financial planning? . . . even leisure? As time and software became available, I learned calculating programs. I learned business planning programs. I became a computer game junkie. I set up a data base using a data base manager program. I even wrote some small programs.

I am reasonably sure that I reached this final stage only after going to work for a large computer company. When I began my new job, I learned new operating systems and new application programs, but I carried into this experience the hard-won semantic knowledge of computing that I had acquired while I was a student. I now work with a very powerful host computer system, and (I'm chagrined to admit) I only scratch the surface of what it can do. My vision of what the computer can do for me continues to grow and change.

[1] Ben Shneiderman, IBM Workshop on Interactive Systems, Kingston, New York, October 14, 1982.

The point is this: I have become a confident computer user and I still read Emily Dickenson and watch Shakespeare on PBS. No sinister transformation has taken place. There is this little thing about my fingers and my brain . . . , but I don't mind this change.

I've had the pleasure of watching at a distance as a friend of mine—a confirmed typewriter addict with twenty years journalism experience—was propelled into the world of technology by having her office go "online." At first, all I heard was complaints. But the last time I talked with her, she made the statement that serves as the conclusion to this part of the chapter. She said, "We humanists really fight using a computer, but once we capitulate, we really go whole hog." (For those of you denied a childhood in the South, that means "we accept it with great enthusiasm.") And you will too.

WORD PROCESSORS AND WRITING CHANGE

The movement of my brain to my fingers has changed forever the way that I generate and refine ideas. For the most part, these changes have been for the better. The computer gives me freedom to:

- *Experiment with alternative versions of an idea.* I can try out different ways of stating the same idea and compare clean versions objectively.
- *Capture and hold fleeting thoughts that otherwise might be lost.* Ideas can be brief and bright as the flare from a match. Writing on a piece of paper is slow, but I can type almost as fast as I can think. I am the Van Cliburn of the computer keyboard. I lose fewer good ideas because the motor skill of writing cannot keep up with the lightning speed of cognition. The computer enhances what writing teachers call "freewriting," brainstorming ideas in writing. The first draft of a paper generated on the computer is a map of my thought processes. It contains the false starts and the dead ends. But it also contains embryonic ideas that might otherwise have been lost.
- *Revise more.* I am not likely to be content with a less than polished product since experimentation is so cheap. With the computer, I feel that I am writing in an experimental environment. I am more than ever involved in the craft of writing, but I have less ego-involvement with my drafts. I am more open to suggestions since change is so easy to make and test.

So even though the computer is a tool, it is a very powerful tool. For me, it has truly become an extension of my thought processes.

WORD PROCESSING PROBLEMS

I don't want all this glowing rhetoric to make you lose sight of the fact that there are some very real problems with using the computer. The first problem, chronologically speaking, is the one I talked about under "Fear and Loathing"

earlier in this chapter, the problem that I've seen referred to as "cyberphobia," the fear of computers—or more generally, the fear of technology. We are all afraid of the new; we are comfortable with things as they are. We have all heard the water-cooler stories about programs that go awry and consume a small fortune in computer time. When I began using the computer, I thought that by touching it, I could destroy an expensive, advanced system. If this were true, then design flaws, not users, would be at fault.

Some kinds of failures are possible and should be anticipated. For example, my home computer has a less-than-adequate cooling system. Once on a hot July afternoon, it locked up and took three hours of composing with it. Once a computer goes down and I have to regenerate an idea, I never seem to be content with the new version. In another instance, I placed a diskette too close to a magnetic paper clip holder and demagnetized it. My hard-learned lesson was to back-up everything from the screen periodically to diskette and to back-up valuable diskettes. New computer users don't know what they can and cannot do with the machines. Cyberphobia is a very real and sometimes not entirely unfounded fear—one that I try to remember exists when I teach my readers about new computer equipment.

A second problem I will refer to as "the finished look." This problem affects us both as writers and as teachers of writing. It's the feeling you used to have in elementary school when you copied a paper over and discovered that you had left out a sentence. But the paper "looked" so nice that you decided to omit the sentence rather than destroy the appearance. The finished look of the paper was too compelling to destroy, even at the cost of making the paper better. Accustomed to seeing clean copy only after typing from messy drafts, I was deceived by the fact that unfinished ideas could take the form of finished ideas. Initially, I placed more value on computer-generated drafts than they deserved simply because printing them out on a letter-quality printer made them look so good. More than once, I "penned" a piece of work furiously at the keyboard, printed it, read it in the heat of the moment, and was convinced that these were the wisest words I'd ever written. When the fever passed the next day, I was always disappointed. The quality of the raw output contributed to the delusion . . . and this was the computer's fault.

No one suffers from this delusion more willingly than student writers. Several studies of teachers grading student essays have found that teachers give higher grades to papers that are neat and easy to read. Computer-generated student papers are both, but they are still student papers. Late at night, grading the 100th essay of the day, we must guard against "the finished look" in student writing, just as we do in our own writing.

A third problem—and this afflicts us as writers more than as teachers of writing—is the ease of self-plagiarism. I have too often gone to a conference several years in succession only to hear the same ideas—if not the same words—expressed over and over by their author. The computer automates this process.

It makes it easy for us to borrow from ourselves. In these days of "publish or perish" qualifications for tenure, it is all too tempting to plagiarize ourselves, to execute a simple word processing copy rather than restate an idea. But ideas evolve through restatement. Each time I write an idea, I embellish it differently. The best statement of an old idea is often the combination of the various ways I have stated the idea before. The computer eliminates the necessity of rethinking—and thereby denies us the benefits of refining.

A fourth problem is that the computer screen is not well suited for editing tasks. For example,

- It cannot represent the relationships between large bodies of text. To get ready for a massive reorganization, I must be able to shuffle pieces of paper like cards in a deck.
- It somehow absorbs errors that leap off the printed page. I admit I don't understand this, but I have seen other writers allude to it also. When I was a student, I was part of a study to determine what effect computer use had on the writing and editing practices of experienced writers.[2] This study confirmed what I had already guessed about myself as a computer writer: I prefer to write using a computer keyboard, but when I edit, I must have pen and paper. The computer screen seems particularly poor for proofreading—but this is why we have dictionary programs.

This fifth problem is not really a problem, but an area of concern. I regret that my whole attitude toward printed information has changed. I have always had a reverence for books, born of spending some time as a librarian and countless hours of my life in libraries. But I have fallen out of love with the written word. Now, when I am doing research, I resent that I cannot do a computer-style locate on a book to find a passage or an idea that I want. I don't want a reference book; I want a data base. I have no patience with beginning at the beginning and ending at the end. This impatience does not affect books I read for pleasure. But when I am *looking* for something from a book, I wish for more efficient ways to get at its information, and I learned this impatience from the immediacy of the computer.

Another aspect of falling out of love with the written word is that my sense of what is "real" has shifted. When I was first learning to use the computer, I distrusted the transience of the computer screen. Words I placed there while generating ideas were not real until I printed them. The computer screen was temporary. The paper was real.

Now, my thought has shifted 180 degrees. Now no work is real until it has become a part of my developing ideas recorded in computer memory. Last

[2] Jean A. Lutz, *A Study of Professional and Experienced Writers Revising and Editing with the Computer and with Pen and Paper*, Dissertation, Rensselaer Polytechnic Institute, 1983.

weekend, for example, I went to work to catch up on some things. I edited a draft with my pen, a task that took several hours. But I came home feeling like I had accomplished nothing because I had not made any changes to the "real thing"—to my computer file. I had moved from thinking that paper is everything to thinking that paper is nothing.

WORD PROCESSORS AND TEACHING WRITING

Since so many teachers of writing have done a better job with this topic than I can do,[3] I will only hit the high points here. But if you are yourself in the process of becoming a computer user, you might not have considered that your students are already accomplished users. Most of them have been companions of the machines since elementary school. When they learn whatever word processing program your university offers, they are carrying with them a body of semantic knowledge greater than ours as teachers. The computer is an acquired companion for us; it was as much a part of their childhood as Bugs Bunny.

The same thing is true of new writers in industry. My programming experience made me an anomaly when I interviewed for my writing job. It is not unusual now for recent job candidates to have written computer music or word processing programs.

Whether your writing "students" are nervous freshmen or new industry writers, you might want to consider:

- Spending a couple of sessions teaching your organization's word processing system. Along with this, teach them to use dictionary and thesaurus programs or other available tools.
- Having students send you "softcopy" (a computer file) of their papers and evaluate them online. When I surveyed a group of experienced users and writers of online information recently, several people mentioned marginalia as a usability tool on the printed page that they wished most to carry to the computer screen.[4] For now, you can add your comments to the beginnings and ends of papers, but watch for clever software programs that will let you indicate your text changes with highlighting or color and add marginalia or even editing marks.
- Using the computer to stimulate freewriting and brainstorming.

[3] Some sources of information about the use of the computer as a tool in teaching writing include: Lillian S. Bridwell, Paula Reed Nancarrow, and Donald Ross, "The Writing Process and the Writing Machine: Current Research on Word Processors Relevant to the Teaching of Composition," in Richard Beach and Lillian S. Bridwell, ed. *New Directions in Composition Research* (New York: The Guilford Press, 1984) and the following volumes of *College Composition and Communication*: May 1983 (Volume 34, Number 2), February 1984 (Volume 35, Number 1), and October 1985 (Volume 36, Number 3).

[4] Annette Bradford and Brenda Rubens, "A Survey of Experienced Users and Writers of Online Information," *Proceedings of the IEEE Professional Communication Society*, October 1985, pp. 269-274.

Also be aware that the computer makes it easier for students to plagiarize each other or to lift ideas from papers written for other classes. To compensate, you might make your assignments more specific and esoteric and spend some class time workshopping writing drafts.

Even with its problems, the computer is a powerful partner both in writing and in teaching writing. It is a tool, but it is much more than a tool. It can become an instrument for generating thought. As you learn to use the computer, anticipate the stages that I described above. Be aware of the problems I warned you about. But now that you are ready to capitulate, anticipate that you will probably go "whole hog." And don't be surprised if your next headache occurs in your thumb.

Contributors

CHARLES E. BECK is an Associate Professor of Research and Communication at the U.S. Naval Academy, and an Associate Lecturer in Management at the College of Engineering Administration, The George Washington University. He holds a Bachelor of Arts in Philosophy from the University of Pittsburgh, a Master of Arts in English from St. Mary's University in San Antonio, and a Ph.D. in Education Administration and Business Administration from the University of Denver. Currently a major in the Air Force, he has taught at the U.S. Air Force Academy, the University of Colorado at Colorado Springs, The Air Force Institute of Technology School of Logistics, Wright State University College of Business, and Anne Arundel Community College in Maryland. His publications have appeared in *Business Horizons, Perceptual and Motor Skills, IEEE Transactions in Professional Communication, Technical Communication, Research in Word Processing Newsletter, Journal of Divorce,* and *Issues in Higher Education.* He has served as a consultant for government agencies and for private engineering firms.

WILLIAM BENZON, currently a free lance writer, has recently completed an overview of imaging technology and computer graphics for Harry Abrams, Inc. entitled *Visualization: We Create the World We See.* Dr. Benzon has taught communication theory and practice at Rensselaer Polytechnic Institute and published articles and technical reports on technical communication, literary theory, and artificial intelligence. For relaxation Dr. Benzon plays jazz and rhythm 'n blues trumpet and paints.

ANNETTE BRADFORD is a Staff Information Developer with International Business Machine Corporation in Kingston, New York. She holds Bachelor's and Master's degrees from Auburn University and a Ph.D. in Communication and Rhetoric from Rensselaer Polytechnic Institute in Troy, New York. Before joining IBM she taught writing at Auburn University, the State University of New

York at Albany, and RPI. As a consultant, she has worked with Copper Publications, SUNY Press, the SUNY Department of Public Administration, and the New York Department of Environmental Conservation among others. After graduating from RPI in 1982, she joined IBM where she designs and writes online information. She has also served as an information planner and has developed and taught classes in online information. Annette has published in *The Journal of Technical Writing and Communication, Technical Communication, The Journal of Basic Writing, The Technical Writing Teacher, Simply Stated, IEEE Transactions on Professional Communications, Journal of the ACM,* and she is a contributor to *Classical Rhetoric and Modern Discourse* and *The Writer's Mind: Writing as a Mode of Thinking.*

TOM BROWNELL, an Assistant Professor at Ferris State College, Big Rapids, Michigan, coordinates the college's Technical Communication Program. He is a columnist for two automotive magazines, a newsletter editor, book author, and active freelance writer who uses word processing to produce all of his publications.

PATRICIA CAERNARVEN-SMITH has used computers to write and design technical publications for eight of the fourteen years she has been in the field. She has worked as a technical editor, technical writer, manager, designer, teacher, and consultant with high technology companies, including several in the Fortune 1000. She has used mainframe computers, minicomputers, and more recently personal computers to design technical reports, brochures, manuals, and training materials. She holds a BA degree from the University of Michigan and an MA from Eastern Michigan University.

CHARLES R. (DICK) FENNO is Associate Professor of Technical Communication at the Air Force Institute of Technology. His publications have appeared in such diverse journals as *Technical Communication, American Speech, Journal of Trauma, ITCC Proceedings*, and in research reports. Currently, he is investigating the relationships between text processing hardware, software, and the tasks of online writing and editing. A frequent speaker at communication workshops, he enjoys collecting and transmitting the practical ideas of other professional communicators.

ROBERT KRULL is an Associate Professor of Communication at Rensselaer Polytechnic Institute. He received his Ph.D. in Communication Research from the University of Wisconsin–Madison. His research interests include computer documentation and interface design, and appeals and effects of computing and television viewing. He has conducted studies of computer use eye motion, effects of online information design on user performance, and physiological responses of television viewers. His research has been supported by the IBM Corporation, the National Institute of Mental Health, The Children's Television Workshop, and state and local governments.

FREDERICK M. O'HARA, JR. is a consultant in technical communication in Oak Ridge, Tennessee, and a senior member of the Society for Technical Communication. He has a B.S. in Chemistry, an M.S. in Technical Writing, and a Ph.D. in Mass Communication and Library Science. He writes, edits, and indexes books, computer documentation, proposals, reports, and policy papers. In 1985 he coauthored *Handbook of U.S. Economic and Financial Indicators* and wrote *Energy Efficiency in Buildings*. He teaches communication for the Office of Graduate Business Programs at the University of Tennessee.

DAVID T. ORR is a former editor for *Tappi Journal* and a past president of the Society for Technical Communication's Atlanta chapter. He now owns a company engaged in freelance writing and editing. Mr. Orr holds a B.S. in Chemistry from the University of Georgia and a B.A. in English from Mercer University.

PHILIP RUBENS, an Associate Professor of Visual and Technical Communication at Rensselaer Polytechnic Institute, studies the graphic and layout features of both printed and electronic texts. His primary interest is the impact of graphic features on user performance and comprehension as well as the ethical implications of new communication technologies. Before joining RPI, he directed technical writing programs at Michigan Technological University and William Rainey Harper Community College. In addition to his academic duties he has served on the Board of Directors for the Society for Technical Communication.

JOHN A. STIBRAVY, an Associate Professor of Technical Communication, School of Logistics, Air Force Institute of Technology, holds a Bachelor of Arts from Davis and Elkins College, West Vriginia, a Masters from Wichita State University in Texas, and a Ph.D. in Higher Education and English from the University of Denver. Currently a major in the Air Force, he has served on combat crews with the Titan II Missile System, and as an instructor for the alternate command post combat crew. While teaching at the U.S. Air Force Academy, he served as course director of the Technical Writing Honors Course. His publications have appeared in *Technical Communication, Research in Word Processing Newsletter, ITCC Proceedings, AFIT Educator*, and *Proceedings of AF Conference in Technology and Training*. He has served as a consultant for government agencies in various parts of the country.

WILL WHEELER first became interested in technical writing after reading *Zen and the Art of Motorcycle Maintenance*. He first worked as a senior technical writer for International Wordsmiths in Vancouver. Since then he has found a variety of stimulating professional challenges, mostly on computer software development. His latest challenge is a contract position in Saudi Arabia where he writes and edits in a research and development department of an international corporation.